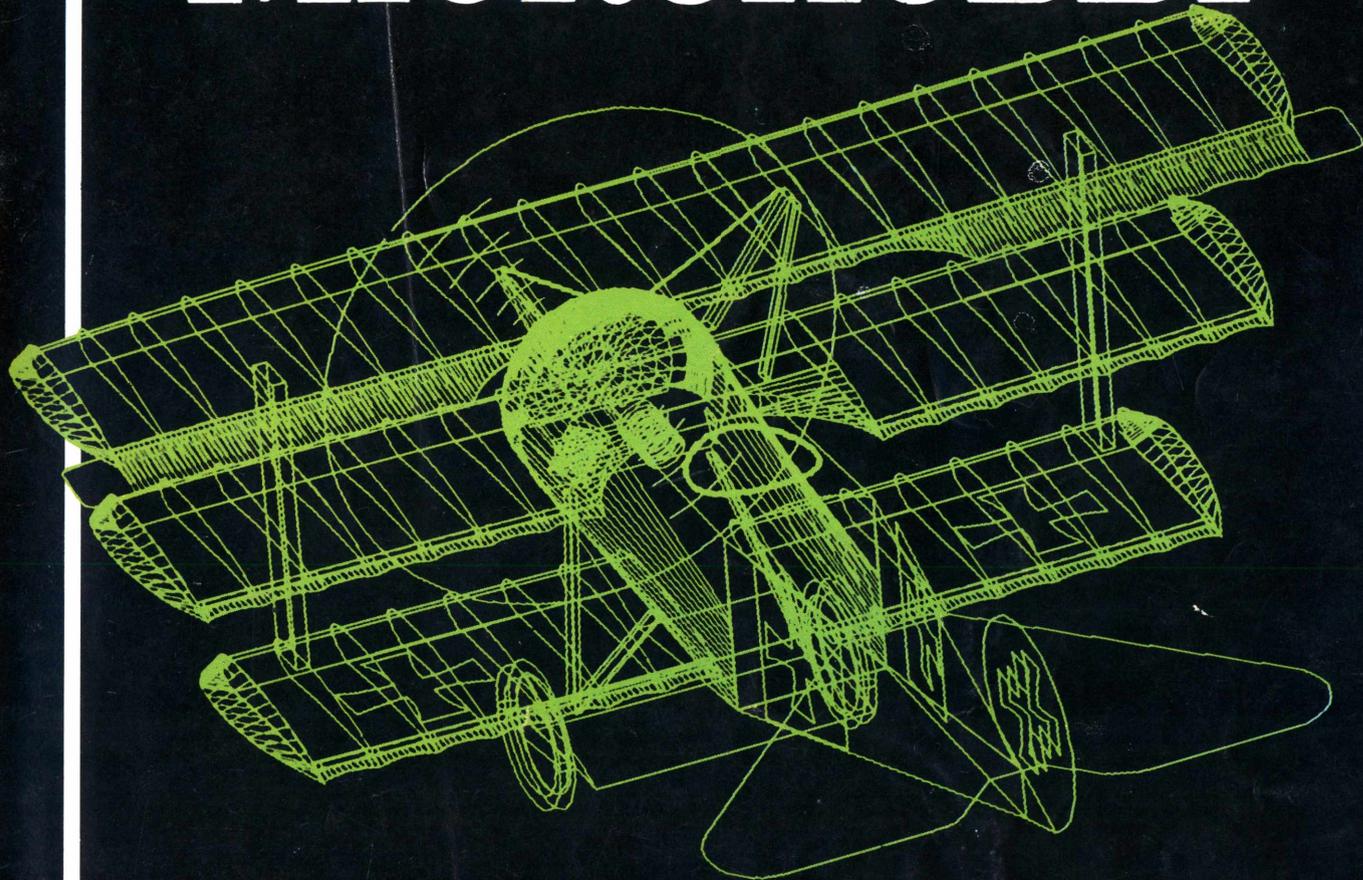
CALCULADORAS

MICROHOBBY

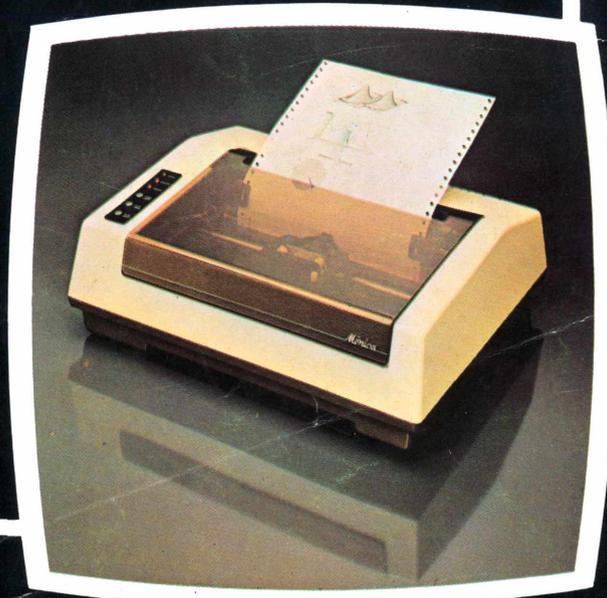
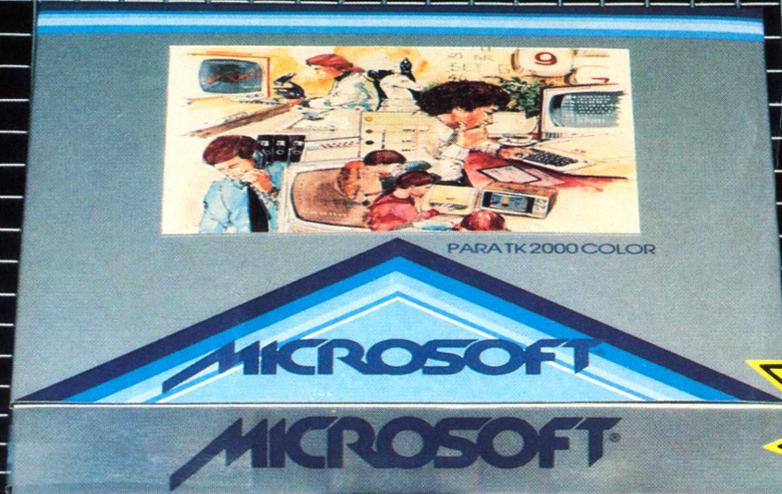
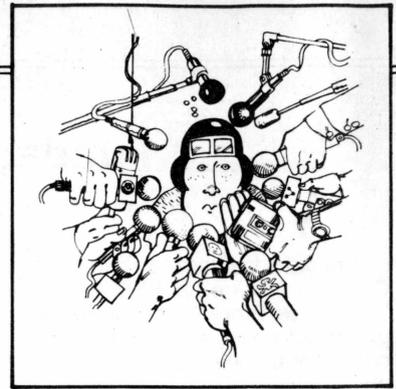
Ano II - Nº 13 - 1984 Cr$ 2.200,00

Para Manaus, Boa Vista, Porto Velho, Rio Branco, Santarém, Altamira, Macapá, Est. de Rondônia (Via Aérea). Cr$ 2.860,00

Reportagem Especial:
COMPUTAÇÃO GRÁFICA

A Volta do Barão Vermelho
TK 83, 85

Analisando:
A IMPRESSORA MONICA

Por Dentro do Apple:
O AMPERSAND

Programa do mês (para TK 2000):
PERSPECTIVA EXATA

Os melhores programas para você.
PARA TK 2000 COLOR
MICROSOFT
MICROSOFT
Garantia integral
MICROSOFT
MICROSOFT
MICROSOFT
MICROSOFT
A Microsoft tem 120 programas em fitas e disquetes à sua disposição. São sistemas aplicativos para acompanhar e agilizar os negócios de sua empresa. E também jogos eletrônicos para você e sua família se divertirem muito. Todos especiais para TK-83, TK-85, TK-2000, Apple II e compatíveis. E todos com a mesma qualidade dos 100.000 programas já vendidos em todo o Brasil.
Procure o revendedor Microsoft mais próximo (se não encontrar os programas Microsoft escreva para a Caixa Postal 54221 - CEP 01000 - S. Paulo-SP). Você encontrará os melhores programas da sua vida.
MICROSOFT
Sempre o melhor programa.

Senac forma unidade de informática

Acompanhando as necessidades do mercado de trabalho e procurando oferecer novas opções, três novas unidades do SENAC (Serviço Nacional de Aprendizagem Comercial) — Informática, Línguas e Propaganda e Promoção — foram inauguradas no último dia três de agosto pelo presidente da Federação do Comércio do Estado de São Paulo, Abram Szajman.

A unidade SENAC Informática pretende oferecer além de cursos, segundo Gerson Roberto Correia, gerente da área, "se tornar um núcleo de estudos e difusão da microinformática, onde profissionais e fabricantes poderão demonstrar o desenvolvimento de seus produtos, através de palestras, seminários, debates e exposições, onde o aluno poderá ver e experimentar as últimas novidades do setor".

Essa unidade tem capacidade para 200 pessoas e conta com 50 microcomputadores, cinco terminais ligados on-line a um Burroughs 6910, quatro salas de aula, quatro laboratórios, um mini-auditório e um show-room para demonstrações. Com todo esse equipamento, desenvolverá quatro tipos de programações: voltado a usuários e público em geral, para profissionais e para alunos de 1º ao 3º graus, com limite de vinte pessoas por turma.

Clube do Micro

Outro projeto é criar um clube do micro, onde ex-alunos poderão continuar usando programas do SENAC ou desenvolver os seus, facilitando um intercâmbio entre profissionais que ainda têm interesse em se aprofundar no assunto.

A unidade também vai oferecer cursos voltados para necessidades profissionais específicas como auditoria, controle de custos, secretariado, engenharia e jornalismo. O objetivo é a capacitação e adaptação dos profissionais dessas áreas a um novo meio. Para Gerson, "o computador daqui a algum tempo se tornará tão indispensável quanto o automóvel, por isso a necessidade de colocar profissionais em contato o mais rápido possível com essa nova ferramenta".

Divididos entre manhã, tarde e noite, os cursos abrangem a linguagem Basic, introdução à microinformática, formação para programador de microcomputadores, linguagem Basic para Apple, Visicalc, programação Cobol. Também foi programado um curso especial para executivos — microcomputadores: contatos imediatos — especialmente preparado para aqueles que buscam conhecimento quanto a utilização e aplicações dos micros. O endereço do SENAC é R. Dr. Vila Nova nº 228 e informações pelo telefone 255-0066 — A.L.M.

Micromirim

A Servimec Processamento de Dados iniciou em junho o "Micromirim", um curso gratuito, especial para crianças de 7 a 14 anos. Durante três horas, aos sábados e domingos, um grupo de doze crianças aprende, sob a orientação de dois instrutores, como operar microcomputadores e softwares.

Floriano Ferreira Júnior, um dos instrutores, conhecido pelas crianças como "tio" Floriano, explica que a introdução da Microinformática nessa idade é importante para que os alunos conheçam mais uma ferramenta de trabalho e aprendam a manejá-la. "Além de desenvolver o raciocínio lógico, o micro põe à disposição uma série de recursos que auxiliam todo o programa escolar, processando um maior volume de informações com maior rapidez", afirma.

Através do "Micromirim" as crianças aprendem sobre os componentes de um microcomputador e desenvolvem pequenos programas em linguagem Logo. Ao mesmo tempo, os pais que acompanham seus filhos ao curso visitam as instalações da Servimec e conhecem desde um equipamento de grande porte a um micro e seus recursos, além de ficarem mais a par da importância da Informática na vida escolar de seus filhos.

Segundo Floriano Ferreira, devido ao sucesso que o curso obteve e pelo entusiasmo demonstrado pelas crianças que, ao fim de cada semana, saem querendo aprender mais sobre o microcomputador, será iniciado em setembro o "BASIC Mirim". Esse novo programa atingirá a faixa etária de 10 a 14 anos. A.L.M.

CALENDÁRIO

3/9 — Microcomputador aplicado no cálculo estrutural — curso destinado a profissionais e estudantes com conhecimento elementar de linguagem BASIC. Informações: Datamicro — Telefone: (021) 274.1042.

3/9 — 5º Congresso Brasileiro de Automática — será realizado em Campina Grande — PR, organizado pela CNPq e outras entidades. Informações: (083) 321.7222, R. 142/442.

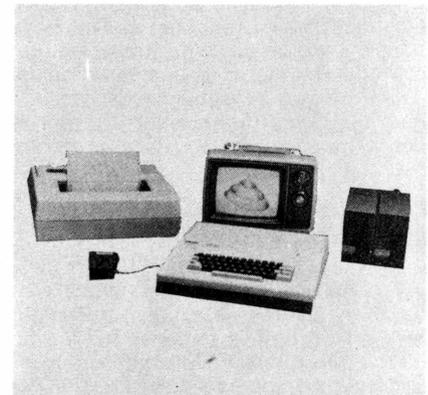
5/9 — Assembler para o Apple — curso com carga horária de 36 horas, realizado pela ORT. Informações: (021) 226.3192.

4/9 — Projeto e Sistemas de Banco de Dados — seminário organizado pela Servimec em São Paulo. Informações: (011) 572.5055.

3, 4, 5 e 6/9 — Simpósio sobre técnicas avançadas do Base II — simpósio realizado pela Compucenter no São Paulo Hilton Hotel. Informações: (011) 255.9662 e 255.5988.

10/9 — Informática em engenharia — curso de especialização para engenheiros de qualquer especialidade organizado pela Univerta — Universidade Aberta no Rio de Janeiro. Informações: (021) 252.0599.

12/9 — Como gerenciar a informação — seminário organizado pela Compucenter no Rio Sheraton Hotel.

Um micro para aplicações gráficas

Com 16 kByte de memória RAM que pode ser expandida para até 64 kBytes, o Codimex-6809 é um microcomputador com vários recursos para aplicações gráficas. Tem capacidade para gerar até 9 cores na tela, opera tanto em alta como em baixa resolução, incorpora instruções que não são encontradas em outros micros além de inúmeros outros recursos como, por exemplo, a capacidade de determinar cores para figuras ou fundo de tela; armazenar desenhos criados no vídeo para posteriores utilização ou duplicação dos mesmos etc. O Codimix-6809, conforme afirmou a empresa, está sendo utilizado por diversos profissionais liberais, escolas secundárias e na UFRGS, nas áreas de pós-graduação em Ciências da Computação. O preço médio do Codimex-6809 está em torno de 140 ORTN's.

Fundação para Livro do Cego inaugura sistema automatizado

A pedido da Fundação para o Livro do Cego no Brasil e do SERPRO (Serviço Federal de Processamento de Dados), um projeto para a utilização de um microcomputador na impressão de livros no sistema Braille foi coordenado e executado durante mais de seis meses pelo professor João Antônio Zuffo, da Escola Politécnica da Universidade de São Paulo. O objetivo era, utilizando-se os recursos já instalados e tecnologia disponíveis no mercado nacional, desenvolver um projeto para aumentar a capacidade produtiva e a qualidade dos títulos editados, através da automação da imprensa Braille.

Com um microcomputador sistema 700 da Prológica e uma impressora P-720, a equipe do Laboratório de Subsistemas Integráveis da USP desenvolveu uma interface que permite o acionamento de uma impressora Braille (o esteriotipo) através do micro, agilizando o trabalho de impressão, feito até agora de forma manual e muito vagarosa. Além da interface, que possibilita o acionamento da impressora, a equipe do professor Zuffo também desenvolveu o software necessário para a codificação do alfabeto e sua transposição para os caracteres Braille.

Após a conclusão do projeto, Dorina Gouvea Nowill, presidente da Fundação para o Livro do Cego no Brasil, inaugurou oficialmente, o equipamento doado pela Prológica à instituição, destacando que o sistema "representa um grande passo na evolução do processo de impressão, pois o livro cujo tempo de produção era de 60 dias, poderá ser feito agora em apenas 42 dias, contribuindo para a melhoria da educação do cego no Brasil".

Os livros impressos são basicamente didáticos, pois existe uma demanda muito grande no país, devido ao número de deficientes visuais, que está em torno de 1,2 milhão. A Fundação trabalha hoje com apenas quatro esteriotipos, alcançando uma produção de vinte títulos por mês ou dois mil volumes, bem abaixo da expectativa do mercado.

"O gato sou eu"

Para Alexandre Roberto Santos Cobianche, engenheiro do Departamento de Produção da Fundação, "o sistema vem beneficiar e agilizar a edição e revisão de textos, reduzindo os custos operacionais e eliminando as provas e revisões após a confecção de matrizes". Segundo ele, o processo atual de transcrição e conversão para o Braille é totalmente manual, provocando erros freqüentes, além de obrigar a revisão das matrizes em três etapas, feita pelos próprios deficientes visuais. Através do novo sistema, essa revisão é feita somente uma vez, diretamente na impressora podendo ser corrigida imediatamente, não provocando uma inutilização da matriz, como ocorria anteriormente.

Quatro operadoras de esteriotipo estão sendo treinadas para trabalhar com o computador, triplicando a capacidade de produção de matrizes da imprensa Braille da Fundação, graças à implantação do processo automatizado. "No entanto", ressalta Alexandre Cobianche, "as outras fases de produção do livro permanecem as mesmas, sem capacidade de absorção correspondente ao nível de produção automatizada". Para ele, a automação só atingirá os objetivos propostos quando todas as fases necessárias à produção de livro em Braille, como montagem e encadernação, estiverem perfeitamente balanceadas, contando com uma equipe mais numerosa.

Todo o programa desenvolvido permite uma futura ampliação do sistema, interligando, no mesmo micro, diversas impressoras em Braille. Alexandre conclui ainda que o projeto é ligar quatro esteriotipos a um mesmo micro, quadruplicando sua capacidade de trabalho. "Tudo isso depende de um auxílio financeiro da comunidade para, além de agilizar os outros departamentos da Fundação, adquirirmos esteriotipos mais rápidos e modernos, acompanhando a tecnologia da Microinformática, afirma.

O livro representa, além de uma maior integração do cego à comunidade, uma das melhores maneiras que ele tem para adquirir conhecimentos não só didáticos como literários para seu próprio lazer. Tanto que o primeiro livro que está inaugurando o sistema automatizado na imprensa da Fundação para o Livro do Cego no Brasil, é justamente o romance "O Gato sou Eu" de Fernando Sabino. A.L.M.

Computer Camp e as férias das crianças

Em Itapecerica da Serra, numa área de 15 mil metros quadrados, o Computer Camp promove todas as férias e feriados, atividades envolvendo o uso de microcomputadores Apple, TK-85 e TK-2000. Nas últimas férias de julho, cerca de 80 crianças em idade entre 8 e 14 anos passaram 12 dias brincando e aprendendo a mexer no computador. Lá, as crianças tiveram acesso a aulas de computação, dadas por dois monitores, à recreação nas quadras de futebol de salão e de campo, natação, participação nas gincanas e no sorteio promovido pela Micromega, onde os vencedores ganharam assinatura da revista Microhobby.

O ensino das linguagens de microcomputação é feito, dividindo as crianças por idade. Os pequenos aprendem LOGO e os maiores, BASIC. O Computer Camp promoverá outras atividades em outubro (entre os dias 10 e 14); novembro (de 14 a 18); dezembro (dias 19 e 20); janeiro (duas turmas) e também em fevereiro. A.L.A.

Programa Microeducação

Instituições de ensino interessadas na aquisição de microcomputadores e na implantação de cursos de Informática especiais para seus alunos têm agora uma alternativa. A Microdigital lançou o Programa de Microeducação, que dá todo apoio às instituições interessadas, auxiliando na implantação de equipamento, literatura especializada e software.

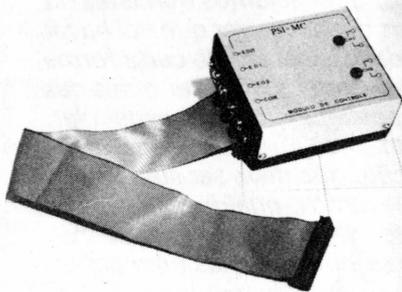
O programa teve início no mês de abril e já conta com numerosos colégios inscritos, tanto de São Paulo como do Interior, inclusive universidades. O objetivo é levar a Informática ao grande público, através de uma maneira simples de aprendê-la.

Segundo Jorge Luiz dos Santos, gerente de Marketing da Microdigital, "os microcomputadores da linha TK vêm se destacando como os computadores pessoais mais adequados àqueles que pretendem iniciar-se nesse campo". Para isso, por meio do Programa Microeducação, as instituições de ensino interessadas na aquisição de micros poderão consultar a Microdigital diretamente e adquirir os computadores por um custo especialmente estipulado.

A empresa fornece aos participantes deste programa amplo apoio técnico e didático. Além do material, oferece cursos para monitores de colégios dando condições para que o micro seja utilizado com o máximo aproveitamento. Jorge dos Santos destaca também que o programa é uma opção para instituições que querem ocupar classes ociosas, não só na época de férias, como no período letivo. A.L.M.

Software para a área financeira

A Monolith 2001 — Eletrônica e Jogos, empresa que comercializa, produz e representa software e hardware nos produtos da linha TK, está colocando no mercado um novo software voltado para aplicações profissionais e domésticas. O programa denomina-se Análise de Investimentos e Financiamentos, ocupa 16 k de memória e pode ser encontrado na Monolith — Rua Gastão do Rego Monteiro, 18 — São Paulo.

Controle automático por micros

PSI-MC é um dispositivo que permite o controle automático de outros aparelhos através dos microcomputadores. O produto, desenvolvido pela PSI-Produtos e Serviços em Informática, possue duas saídas com potência de 60 W cada e permite acionamento de aparelhos como motores, alarmes, brinquedos eletrônicos além de aplicações profissionais na área de controle de processos industriais, laboratórios e bancadas de teste.

Software americano representado pela Compucenter

Os softwares criados pela Microsoft Corporation, como por exemplo o Project e o Chart, já podem ser comprados no Brasil, na Compucenter — Micro Informática. Na compra destes softwares, o usuário recebe um kit composto de disquetes, manual de instruções, certificado de garantia, registro e treinamento fornecido pela empresa.

Um outro software da empresa americana que pode ser adquirido na Compucenter é o Multiplan, nas versões CP/M e Apple-DOS. O Multiplan é a planilha eletrônica mais usada nos Estados Unidos e permite processar os dados pelos seus próprios nomes (vendas, custos, lucros etc.) do mesmo modo como o cérebro humano os processa.

O agrimensor e a planilha de cálculo analítico

O programa Agrimensor, desenvolvido por uma empresa do Rio Grande do Sul permite a redução do tempo de cálculo da planilha que fornece ao técnico a área total da terra que ele deseja medir. A elaboração desta planilha é muito demorada devido à quantidade de números e à complexidade das derivações. Com o Programa Agrimensor, reduz-se o tempo de cálculo permitindo, por exemplo, o cálculo de uma planilha de 34 estações num tempo de aproximadamente 40 segundos. O preço médio do programa é de 6 ORTNs e pode ser adquirido pela CP 289, Santa Rosa — RS.

Videotexto via micros

Os usuários dos microcomputadores da linha Apple já podem ter acesso aos serviços fornecidos pelo Videotexto Telesp através de sistemas específicos desenvolvidos pela Video Soft, uma empresa fornecedora de um destes serviços. Estes sistemas incluem a geração automática de páginas, o acesso ao banco de dados (através da simulação do sistema "fiet" francês, de composição gráfica), a criação de grafismos e animações de tela, a consulta ao videotexto através dos micros e uma linguagem em português — a VXLAN — que permite a programação de grafismos.

Segundo Sergio Cruz Navega, diretor da empresa, a idéia de desenvolver estes sistemas surgiu com a preocupação de agilizar a emissão de exames laboratoriais do CADI — Centro Auxiliar de Diagnóstico — (uma das divisões da empresa que fornece resultados de exames laboratoriais a médicos, clínicas e hospitais através do videotexto). Esta preocupação levou a empresa às pesquisas com o objetivo de simular o sistema "fiet", usado pelo videotexto brasileiro, no micro para que este executasse a entrada de dados, enviando a página formatada na tela com as cores prédeterminadas diretamente, via modem, para o computador central da Telesp, através de uma linha paralela de videotexto.

Para implementar estes sistemas a empresa já entrou em contato com fabricantes nacionais de compatíveis com o Apple, faltando apenas acertos finais.

Uma destas empresas (a qual Sérgio Cruz fez questão de omitir) pretende colocar programa embutido na máquina em Eprom.

O acesso ao videotexto através de micros é possível, segundo o representante da empresa, graças à uma interface (placa RS 232), ao software (cuja representação ficará por conta da Video Soft) e as adaptações feitas no hardware dos micros Apple. Sérgio Cruz acrescentou que a Video Soft já dispõe de software que permitem a gravação de páginas em disquetes, o armazenamento do tempo de páginas na tela para uma consulta rápida e a ligação de impressoras em alta resolução para impressão de páginas. Esta ligação abre, para Cruz, espaço para aplicações profissionais, atingindo empresas que necessitam transmitir informações rapidamente para suas filiais e também profissionais liberais como por exemplo, advogados e engenheiros. A.L.A.

Micros pessoais nos bancos

Uma das palestras apresentada durante a realização do X Congresso Latino Americano de Automação Bancária em São Paulo, trouxe uma informação inovadora para o usuário que utiliza micros pessoais em aplicações profissionais: a notícia de que a Burroughs americana já comercializa um microcomputador com um sistema operacional — o BTOS — que permite o trabalho de 16 micros no mesmo arquivo, ao mesmo tempo. Talvez por este motivo e por ser a empresa um dos expositores presentes a feira, que ocorreu em paralelo ao Congresso, os organizadores trouxeram para falar sobre o tema Micro Pessoal nos Bancos, Richard E. Ponton, vicepresidente de Marketing da Burroughs americana.

Muitos grandes bancos segundo Ricard E. Ponton, estabeleceram um depósito de computadores internos que geralmente têm dois ou três micros, além de terminais e software que permitem ao usuário alcançar o estoque e receber aviso para aplicações adequadas do software e do hardware. Com a interferência de departamentos que ele denomina de MIS (provavelmente centros ou gerência de processamento de dados) incorporando os micros em suas aplicações, as agências bancárias passaram a requerer "data bases" locais para relações de clientes mais eficientes e as aplicações usuais passaram a ser escritas para a nova tecnologia. O interesse pelos micros pode atacar, conforme acrescentou, um problema grave enfrentado pelos bancos, pois, enquanto estes dispendem uma alta percentagem de seus orçamentos de operação em equipamentos computadorizados e serviços e suas operações financeiras são também um dos negócios mais intensivos no mundo, os índices de crescimento na produtividade do funcionário está entre os mais baixos de todas as indústrias.

Atualmente, grandes organizações americanas estão sanando este tipo de problema com a opção por micros que têm a mesma capacidade de minis computadores. Um exemplo, são os micros B25 da Burroughs que foi desenvolvido há aproximadamente dois anos atrás. Conforme afirmou Richard E. Ponton, a empresa desenvolveu um sistema operacional único até o momento no mercado americano, denominado BTOS. O micro, acrescentou, atende as mesmas necessidades dos sistemas de computadores de maior porte. Permite aplicações multiusuárias, possue recursos que facilitam a migração da informação para usuários de sistemas de equipamentos menores (MS/DOS e CP/M) e tem uma capacidade de memória de 1 Megabyte e 40 Megabyte de armazenamento em discos. A.L.A.

EDITORIAL

No presente momento o Brasil está aguardando mudanças em diversos setores: políticos, sociais, econômicos e tecnológicos. Existem expectativas otimistas e pessimistas em qualquer uma das alternativas que se apresentam em cada um destes setores, mas, quer sejamos otimistas ou pessimistas, não podemos deixar de ver que vai haver mudanças e que todas elas estão de certa forma interligadas. Não podemos prever quais das alternativas serão escolhidas, apenas especular. entretanto, existe algo, que independente de qualquer especulação, que deve ser sustentado como princípio. Devemos, qualquer que seja a alternativa, defender a tecnologia nacional.

Isto não significa apenas uma defesa intransigente da reserva de mercado. É claro que esta defesa deve continuar.

Entretanto, devemos defender a tecnologia nacional ainda que a reserva de mercado caia. E isto é bem mais difícil de fazer.

A impressão que se tem é que, se a reserva cair, seremos engolidos por enormes dinossauros. Seremos, se não estivermos preparados. Eu acredito que esta preparação já deve ser feita por todos executivos de empresas nacionais, sem que isto indique pessimismo quanto ao futuro.

Outra coisa a se notar é que nos Estados Unidos, país da vanguarda da tecnologia Informática, os novos computadores já envolvem tecnologia que ainda não dispomos, pois estão baseadas em circuitos dedicados, que tornam seus modelos de microcomputadores mais baratos, avançados e com uma tecnologia nova, difícil de ser assimilada.

Diante disto, fazemos uma pergunta que consideramos importante: aonde está a nossa indústria de microeletrônica? Nos laboratórios de pesquisas? Em projetos de SEI? Em alguma empresa? Só nos colocaremos em igual de condições para uma "briga" tecnológica com empresas do porte da Apple, IBM, DATA General e outras se tivermos chips dedicados (ao lado dos não-dedicados) disponíveis no mercado, produzidos por empresas nacionais. Quem se habilita?

****

Este mês adotamos em definitivo o "Barão Vermelho", herói alemão da I Guerra Mundial, como nosso personagem.

A rigor ele nos tem acompanhado desde o número 0, aparecendo ou sendo mencionado de quando em quando. A partir deste número ele será um personagem constante, abrindo todas as seções e participando dos jogos onde ele for convocado. A razão desta escolha é simples: além de ser há tempos o personagem adotado por nós (pelo menos a nível interno), ele simboliza a coragem e a nobreza que todos (nós da redação e vocês leitores) gostariam de ter, uma coragem e nobreza reconhecida até pelos seus adversários, o que dificilmente ocorre num estado de guerra.

Bem vindo, Otto Von Richthofen.

Índice

Capa
A capa deste número foi criada por Roberto Bertini Renzetti, elaborada em um equipamento Integraph por Nilson Lobo Pinto e fotografada por Israel Teixeira.

Agradecemos à Intergraph pelo uso dos equipamentos, que gentilmente nos cederam, para a elaboração desta capa.

Expediente

EDITOR
Álvaro A.L. Domingues
JORNALISTA RESPONSÁVEL
Ana Lúcia de Alcântara — Mt. 14.495
EDIÇÃO DE NOTÍCIAS
Ana Lúcia de Alcântara
REDAÇÃO
Colaboradores: Ana Lúcia Mahlmeister
ASSESSORIA TÉCNICA
Flavio Rossini
Wilson José Tucci
ANÁLISE E REDAÇÃO
Caio Marques Bulhões
Rogéria Gomes da Silva (secretária)
Vivian Bernardo
PRODUÇÃO GRÁFICA E DIAGRAMAÇÃO
José Carlos Sarkis

CORRESPONDENTES
New York — Flavio Rossini
PUBLICIDADE
Aurio José Mosolino (supervisor)
Eduardo Garcia de Souza
ASSINATURAS
Marcia Regina Dominiqui
Marcos Lorenzi
CIRCULAÇÃO
José Aparecido Bueno
GERÊNCIA GERAL
Angel D. Zaccaro Conesa
Dijalma Peinado
DISTRIBUIÇÃO
Fernando Chinaglia Distribuidora S/A
COMPOSIÇÃO E FOTOLITOS
Ponto Reproduções Gráficas Ltda.
IMPRESSÃO E ACABAMENTO
Editora Parma Ltda.

MICROHOBBY é editada mensalmente por MICROMEGA PUBLICAÇÕES E MATERIAL DIDÁTICO LTDA., INPI 2992 Livro A
Endereço para correspondência:
Caixa Postal 54096 —
CEP 01296 — São Paulo, SP.
Para solicitar assinaturas (12 números) envie cheque nominal à MICROMEGA P.M.D. LTDA., no valor de Cr$ 22.000,00.
Tiragem desta edição: 30.000 exemplares
NÚMERO 13

CARTAS

TK 2000

(. . .) Tendo adquirido recentemente o TK 2000, ainda estou dando os primeiros passos. E creio ser este o único motivo da minha falta de conhecimento sobre o assunto, principalmente em relação ao mapa de memória do TK 2000 e do Apple, apresentados em artigo recente da Microhobby.

Gostaria de fazer perguntas objetivas, mas o assunto é extenso e, portanto, o mais certo seria que me indicassem livros que se referissem a assuntos mais complicados, que são as linguagens de máquina, o traçado de gráficos e jogos, bem como o conhecimento da estrutura de memória do TK 2000. Gostaria de saber também quais são as revistas especializadas que tratam do assuntos exclusivos da linha Apple. Isto porque não encontrei nenhum, apenas a nova seção que se abre agora na revista Microhobby, a qual devo agradecer.

Se possível, gostaria de explicações a respeito dos códigos que marcam o início e o fim das seções do mapa de memória (ex: $1FFF) uma vez que estou acostumado a ver apenas números como indicadores do endereço de memória. Para finalizar, o que vem a ser página de vídeo?

André Luis Muschioni
Belo Horizonte, MG

Caro André,

Por ser o TK 2000 uma máquina nova no mercado, ainda não existe uma literatura especializada sobre ele, a não ser seus manuais e o que estamos mostrando nas páginas de Microhobby. Todavia, é intenção da Micromega fazer um ou mais livros sobre o assunto.

As revistas especializadas, que memcionamos nas revistas 10 e 11, sobre o Apple são extrangeiras e você pode encontrá-las em livrarias que comercializam revistas e livros importados. Entre elas, você pode encontrar duas, que consideramos boas, a In Cider e a Nibble, que fornecem bastante informações sobre o Apple.

Os códigos que aparecem estão em hexadecimal e, por este motivo, existem as letras no seu código. Na tabela a seguir mostramos como converter números hexadecimais em decimais. O símbolo $, neste caso, significa que um número está na base 16 e não na base 10.

base 10	base 16	base 10	base 16
0	0	8	8
1	1	9	9
2	2	10	A
3	3	11	B
4	4	12	C
5	5	13	D
6	6	14	E
7	7	15	F

Merge par ao TK 2000

Sou assinante e assíduo leitor da revista Microhobby. No último número (11) foi publicada uma rotina que implementa o comando Merge nos micros compatíveis com o TK 85, que acredito ser de muito valia para os usuários destes microcomputadores. Possui um CP 200 por mais de um ano e senti muito a falta de um comando deste tipo.

Atualmente sou possuidor de um TK 2000 Color e gostaria de sugerir a publicação de uma rotina de Merge para este micro, pois acredito que muitos usuários do TK 2000 também sentem a falta deste comando.

Existem atualmente no mercado nacional vários compiladores para outras linguagens que não o BASIC (Forth, Pascal e outras) para computadores compatíveis como Apple. Gostaria de saber se existem alguns destes compiladores disponíveis também para o TK 2000, já que o mesmo não é totalmente compatível com o Apple.

Salustiano Campelo Sampaio
Fortaleza, CE

Caro Salustiano,

Suas sugestões são válidades e procuraremos, dentro do possível, atendê-las. Quanto aos compiladores, no momento desconhecemos qualquer empresa que os esteja fabricando e comercializando este tipo de software. Fica aqui registrado como sugestão aos fabricantes de software o desenvolvimento de compiladores para o TK 2000.

Hardware do TK 85

Considerando o número cada vez maior de usuários de micros compatíveis com o TK, gostaria de questionar por que, até o número 11 da Microhobby, não foi abordado nenhuma aplicação em hardware para micros TKs 83/85.

Outrossim, quando forem usadas listagens de programas feitas pela impressora, deveria ser especificado no texto quais os símbolos gráficos utilizados em cada linha de programa.

Desejaria ainda sugerir a abordagem, em edições posteriores, dos seguintes assuntos:

1º Linguagem FORTH — Um curso sobre esta linguagem, uma vez que ela é bastante rápida e economiza memória e já temos no Brasil um compilador para o TK (Estou falando do TIG-FORTH, fabricado pela TIGRE);

2º Medidor de temperatura — usando conector para expansão do TK como entrada de dados ligado a um coversor A/D, que transformaria a tensão gerada por um transdutor em dados.

3º Discador automático de telefone.

4º Transformada de Fourier — um programa que forneça o conteúdo harmônico d eum sinal, a partir de suas amostras;

5º Light pen.

6º Interfaces RS 232 e Centronics — Como usá-las no TK?

Caro Robson,

O hardware será objeto de uma seção a ele dedicada, seção esta que já existe, mas tem aparecido apenas esporadicamente, apresentando pequenos projetos e dicas úteis. Esta seção será ampliada com projetos mais elaborados de melhor nível, destinada aos leitores com algum conhecimento de eletrônica.

Suas sugestões são interessantes e merecerão atenção de nossa parte. Pesquisaremos sobre a viabilidade e o interesse de tais assuntos.

TK 85 abandonado? (I)

Microhobby tem altíssimo nível nas matérias que publica e seria ótimo que se a Microhobby fosse "A REVISTA DOS USUÁRIOS DO TK", uma vez que para os "outros" computadores temos mais de 20 revistas especializadas, "que são também publicações de alto nível só que discriminam o pequeno TK".

Leonardo Chiarelli Linhares

Caro Leonardo,

Quando ampliamos a linha editorial tomamos algumas precauções:

a) não deixamos de lado o TK 83/85. Ao contrário, começamos a nos preocupar com aplicações mais sérias que não os joguinhos (se bem que eles continuarão a aparecer nas nossas páginas).

b) aumentamos o número de páginas.

c) diminuímos o tamanho das letras em algumas seções.

d) estamos alterando a diagramação buscando uma melhor distribuição dos assuntos e anúncios.

e) aumentamos a parte jornalística, trazendo maior número de informações relevantes ao leitor.

Todas estas modificações abriram, ampliaram e melhoraram a revista. Não concordamos que existem mais de 20 publicações destinadas a outros computadores, nem que os TKs estejam desprezados, por nós ou por outros. Dê uma olhada nas bancas de jornais: você verá, além da nossa, apenas sete ou oito publicações nacionais, nem todas dedicadas ao hobista ou possuidores de computadores pessoais. Não entraremos aqui em comparações, pois seria anti-ético de nossa parte, mas, mesmo com a nossa ampliação, acreditamos que a porcentagem de páginas destinadas aos micros compatíveis com o TK é bastante significativa, bem como o nível de qualidade dos assuntos abordados, em qualquer das linhas.

TK 85 abandonado? (II)

Quero, em primeiro lugar iniciar esta carta com um elogio. Envio meu reconhecimento da profunda melhoria que havia a cada número da Microhobby. É ótima a idéia de ampliar a gama de assuntos principalmente a colocação de seções do tipo Por Dentro do Apple e do curso de linguagem de máquina. Mas na minha opinião a melhora não para aí. É notadamente proficua a atenção dispensada aos vários níveis de conhecimento dos leitores desta revista.

Por meio dela tomei conhecimento de vários macetes e aplicações computacionais. Discordo, inclusive da opinião de vários leitores que tem escrito contra a ampliação da revista em relação a outros computadores que não sejam compatíveis ao TKs 83 e 85. Eu, por exemplo, iniciei num TK82-C e agora possuo um compatível com o Apple e não abandonei o TK.

Marcelo Araujo Brandão
Alfemas, MG

Caro Marcelo,

Agradecemos seus elogios e sua manifestação de apoio à nossa linha editorial. Esperamos que nossos leitores compreendam nossa posição, como você o fez. Afinal, continuamos a prestar bons serviços aos possuidores de computadores, principalmente dos TKs.

DESGRILANDO

INKEY$, SAVET e o TK 2000

Em primeiro lugar, dou-lhes parabéns pela excelente qualidade do periódico que editam.

Tenho verificado que as respostas às dúvidas dos leitores são sempre bem explanadas.

Agradeceria, portanto, se analisassem e respondessem as perguntas que estou lhes encaminhando.

Lido com microcomputadores há mais de um ano e recentemente adquiri um TK 2000. Tenho, porém, alguns problemas com o manuseio correto de algumas instruções deste computador.

Dirijo-lhes aqui duas questões:

Questão 1: Desejava gravar um arquivo de dados em fita cassete. Acrescentei no meu programa uma linha contendo o comando SAVET "nome". Aconteceu o seguinte:

a) Ao listar o programa, a linha em questão foi impressa como

300 SAVE T "nome"

ou seja, com o T separado da instrução.

b) Ao rodar o programa com dados que foram introduzidos, por uma opção existente no programa, chamei a linha 300, para gravar o programa mais os dados. Após alguns segundos (como se estivesse ocorrendo a gravação), o micro apresentou a mensagem de erro de sintaxe na linha 300.

c) Posteriormente, constatei que, se gravasse o programa, após a sua execução, usando o comando SAVET "Nome" no modo direto, o computador gravava somente o programa, mas não salvava os dados. O mesmo ocorria com o comando SAVEA (formato Apple).

Pergunta: Qual o processo para gravar os dados de um programa, seja eles variáveis simples ou matrizes (numéricas ou string)? E como recuperá-los?

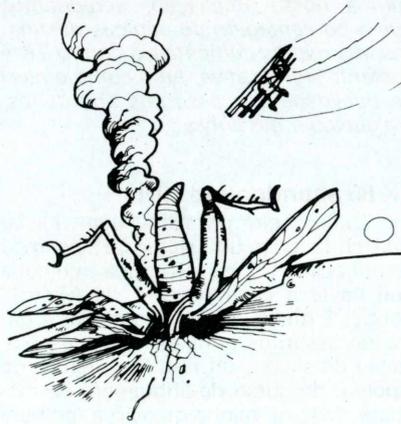
Questão 2: Tentei simular um INKEY$ no TK 2000. Por um manual do Apple soube que PEEK (−16384) lê a tecla pressionada. Este endereço não funcionou no TK 2000.

O programa testado foi:

```
10 PRINT PEEK(−16384): GOTO 10
```

No Apple, a cada tecla pressionada, o número do seu código será mostrado na tela. O mesmo não acontece no TK 2000.

Pergunta: Qual o endereço no TK 2000 que lê a tecla pressionada?

Albino José de Iório
Juiz de Fora — Minas Gerais

Agradecemos os elogios que nos fez, Albino. Em relação às dúvidas, temos sempre feito o possível para saná-las, procurando sempre um melhor atendimento de nossos leitores.

As respostas que temos para as suas dúvidas são:

Questão 1: *A instrução SAVE do TK 2000 (Nas formas SAVEA e SAVET) salvam apenas o programa, ignorando suas variáveis.*

Devido à sua estrutura de trabalho, o TK 2000 pode gravar e ler apenas matrizes numéricas, através das instruções STORE e RECALL, que são explicadas em detalhes no manual do equipamento.

Uma maneira de gravar e ler strings é colocar em seu programa uma rotina que, através da função ASC, converta a string para uma matriz numérica e, por meio de função CHR$ faça a operação inversa no momento da leitura.

Obs.: *O comando SAVET só pode ser usado no modo imediato, não podendo ser usado em linhas de programas.*

Questão 2: *O Teclado do TK 2000 funciona de maneira diferente do teclado do Apple. Enquanto no Apple o teclado é "semi-inteligente" no TK 2000 ele é matricial. Isto significa que, antes de um programa em BASIC ler de uma posição de memória o código de uma tecla pressionada, a CPU deverá afetar a varredura da matriz do teclado, gerar o código correspondente à tecla que estiver sendo pressionada e armazená-lo na memória. Para fazer isso, podemos usar esta pequena rotina em Assembly:*

```
JSR $F043  ; Executa a varredura do teclado e gera código
STA $3A    ; Armazena o código da tecla lida
RST        ; Retorna
```

Você deverá, logo no início do seu programa introduzir esta rotina na memória. Isto pode ser feito em apenas uma linha de programa:

```
DATA 32, 67, 240, 133, 58, 96: FOR F = 768 TO 773: READ X: POKE F,X: NEXT.
```

Após a rotina em assembly ter sido introduzida na memória, para silar o INKEY$ basta colocar em seu programa, as seguintes instruções:

```
CALL 768: REM EXECUTA A ROTINA EM ASSEMBLY
X = PEEK (58): REM RECUPERA O CÓDIGO DA TECLA
```

Desta forma, o conteúdo da variável X será o código da tecla ou zero, se nenhuma tecla for pressionada.

GET X$

Para alguns usos, podemos ainda usar a função GET X$, que equivale a

```
100 A$ = INKEY$: IF A$ = "" THEN 100
```

Ou seja, GET aguarda que uma tecla seja pressionada, enquanto que o INKEY$ executa a próxima instrução se nenhuma tecla tiver sido pressionada.

Merge no TK 83/85

Por diversas vezes li em várias revistas, inclusive na Microhobby e no Manual do TK 85, para ir guardando aos poucos os programas muito longos. Fiquei muito contente ao saber disso, mas quando tentei pela primeira vez, na digitação do programa "Bate Coração", isto não funcionou. Eu dividi o programa em três partes. Quando carreguei a primeira parte, tudo bem, mas quando carreguei a segunda, a primeira desapareceu, o mesmo ocorrendo com a segunda; quando carreguei a terceira.

Gostaria de saber se alguma tática especial para carregar programas. (. . .)

Ranieri Slemer
São Paulo — SP

Você não entendeu o que foi dito, Ranieri. Quando fala-se em gravar programas em partes, o procedimento é o seguinte:

a) digita-se uma parte do programa

b) grava-se este programa. Guarda-se para uso posterior.

c) carrega-se este programa novamente quando tivermos intenção de continuar a digitação.

d) digita-se mais uma parte logo em seguida.

e) grava-se todo o programa.

f) repete-se o processo até que o programa esteja todo digitado.

Este procedimento é útil quando se tem um programa muito extenso e deseja-se digitá-lo. É extremamente desagradável, após ter-se digitado quase todo o programa, perdê-lo por motivo qualquer, como, por exemplo, uma queda na tensão.

Da maneira como você fez, somente se tivermos um Merge na RAMTOP é que conseguiremos fazer a junção de dois programas gravados em fita, como você nos descreve. Na revista Microhobby número 11 publicamos um programa com esta finalidade e, nas revistas 12 e 13, explicamos detalhadamente o que acontece quando se faz este tipo de operação.

As funções secretas no D-8000

Você é possuidor de um D-800, 8001 ou 8002 e gostaria que ele tivesse mais algumas funções? Se você tiver sorte, existem algumas funções extras não divulgadas em seu manual e que são bastante úteis.

Toshinobu Ishida

A família de micros D-8000 foi montada pela Dismac durante alguns anos, até que a sua produção foi repentinamente suspensa. Apesar de haver, hoje em dia, diversos micros mais novos e poderosos, muitos ainda o utilizam com todo carinho.

Só que muitos possuidores do D-8000 sentem uma pontinha de inveja ao ver os modernos TRS-compatíveis, pois estes possuem letras minúsculas, cursor piscante, som, maior recurso gráfico, katakana, teclas auto-repetitivas . . .

Mas não desanime. Se o seu micro for da terceira geração, terá uma grata surpresa ao ler este artigo.

A terceira geração da família D-8000 possui uma ROM adicional em relação aos mais antigos. Nela estão gravados rotinas em linguagem de máquina que implementam:

a) Obtenção de letras minúsculas.

b) Obtenção de caracteres especiais em algumas teclas.

c) Teclas auto-repetitivas.

d) Cursor piscante.

e) Renumerador de linhas de programa.

Para ler esta rotina digite SYSTEM e aperte NEW LINE. Ao obter o sinal *?, responda digitando /12288. Aperte NEW LINE. Se o seu micro responder NEW KEYBOARD ROUTINE ENABLE, ótimo! Caso isto não ocorra, infelizmente ele não é equipado com este recurso adicional.

As letras minúsculas são obtidas pressionando simultaneamente a tecla SHIFT e a letra correspondente. Pelo mesmo procedimento, obtém-se caracteres especiais em mais algumas teclas.

Mantenha pressionada uma tecla. Você reparou que as letras estão se repetindo mas antes da repetição, há um intervalo de tempo de aproximadamente um segundo? Excelente! Isto contorna o crônico problema do teclado do seu micro que tem o péssimo hábito de repetir letras, mesmo quando as teclas são pressionadas levemente.

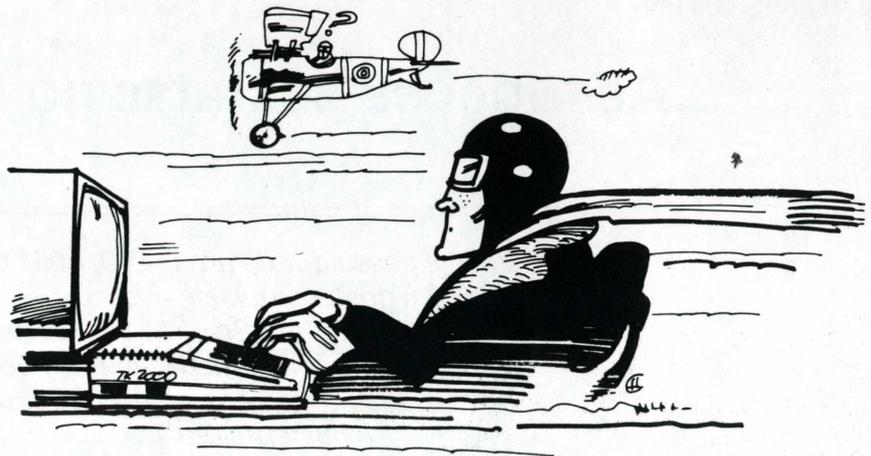
O cursor piscante pode ser desativado ou reativado através de uma mesma operação. Ou seja, pressionando simultaneamente SHIFT e BREAK.

E por último o sensacional renumerador de programas que nenhum micro da sua classe possui na sua configuração mais simples. A sintaxe do comando é idêntico ao do AUTO.

RE 100,10

A instrução acima renumera o programa de tal modo que na nova forma a numeração inicie no 100 e o incremento seja de 10. Apesar de renumerar também todos os GOTOs, GOSUBs, suas composições e IFs, não renumera o RESUME.

Por último um desafio. Siga o procedimento anterior, mas ao invés de digitar /12288, experimente digitar /13127 e você verá os conteúdos dos registros do Z-80. E mais, algumas teclas permitem acessar rotinas especiais tais como R, M e G. Para voltar ao BASIC sem desligar o micro aperte a tecla B.

Rotação

Coordenação:
Prof. Wilson José Tucci/Domus Informática
Colaborações de
Luiz C. N. Szente
José Eduardo Moreira
Daniel R. Falconer

Neste artigo, nós vamos continuar a explicar a formação de figuras animadas na tela.

Iremos apresentar agora a rotação de figuras.

Para tanto é necessário uma pequena introdução matemática do que consiste a rotação de um ponto.

Inicialmente, seja o sistema de coordenadas X, Y e o ponto P (figura 1). Para nos referenciarmos ao ponto P, nós nos referimos à componente do eixo X, A e a componente ao eixo Y, B ou seja $P = (A, B)_{x,y}$.

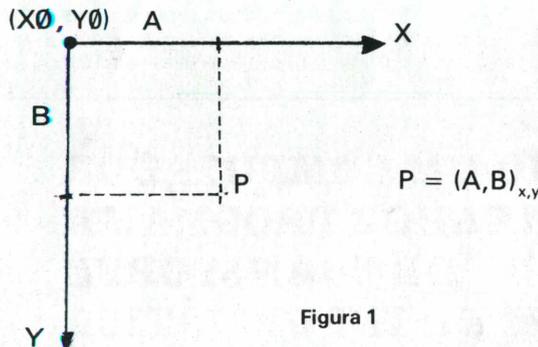



Figura 1

Imagine agora que giremos a folha de papel de N graus em relação ao ponto XØ, YØ. Desta forma, teremos um novo sistema de coordenadas X', Y' (figura 2). O ponto P' ainda pode ser referido a esse sistema de coordenadas pelos componentes A, B ou seja $P' = (A, B)_{x',y'}$.

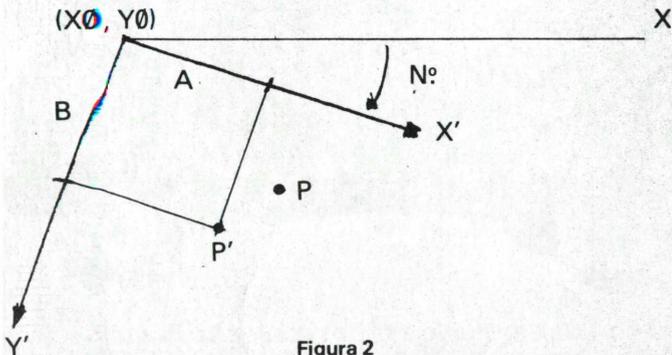



Figura 2

O ponto XØ, YØ, como permanece fixo, é denominado *centro de rotação.* Observe que o ponto P se movimenta para P'. Para colocarmos tal ponto no vídeo, nós temos que nos referenciar ao esquema de coordenadas X, Y, ou seja P' (XP', $YP')_{x,y}$.

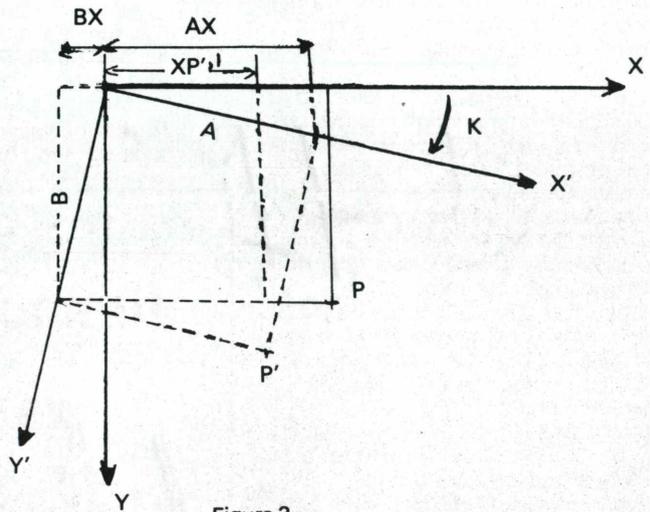



Figura 3

Para obtermos tais componentes, (XP', YP'), temos que "projetar" as componentes A, B do sistema X'Y' para o sistema XY.

Projetar uma componente significa multiplicarmos a componente de um sistema pelo cosseno do ângulo formado entre ela e o eixo.

Assim para obtermos a componente do eixo X, (XP'), teremos que projetar a componente A e a componente B para o eixo X (Pense: o ponto P' é formado por A e B. Para que se tenha a formação XP', tem-se que projetar essas formações para o eixo X).

Logo: XP = AX − BX
onde: AX = A COSK
BY = B COS(90 − K) = BSIN(K)

Para obtermos YP', o procedimento é análogo, ou seja:
YP' = AY + BY
YP' − ASSIN(K) + B cos(K)

BY = B cosK
AY = A cos(9Ø−K) = ASIN(K)

Imagine agora que se tenha um quadrado e desejamos aplicar um giro de 30º em relação a seu centro (figura 4).

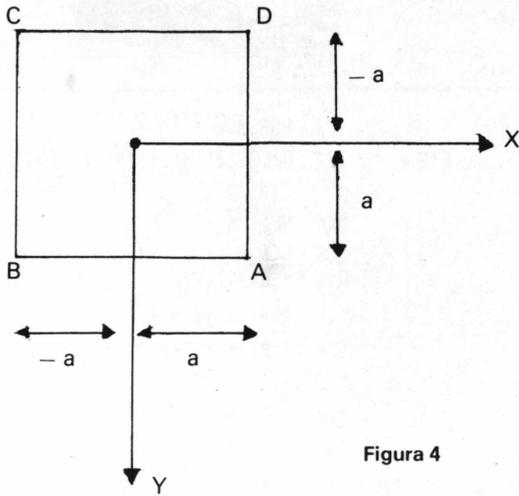



Figura 4

As coordenadas são: A (a,a)
B (– a,a)
C (– a, – a)
D (a, – a)

e passarão a ser (figura 5):

A' (a cosK – a senK)
B' (– a cosK – (– a) senK)
C' (– a cosK – a senK)
D' (a cosK – (– a) senK)

Figura 5

Fácil, não?

O programa

```
0  REM ØEXPLORANDO O TK2000
     ROTACAO DE FIGURAS
        1984
10  HGR : HCOLOR= 7
20  PRINT "COLOQUE AS COORD. REL.
     DA FIG. "
30  DIM XI(20),YI(20),XF(20),YF(2
     0)
40  DIM X1(20),Y1(20),X2(20),Y2(2
     0)
50  DIM X3(20),Y3(20),X4(20),Y4(2
     0)
60  DIM C(2,2)
70  PRINT "ENTRE COM AS COORD.  >
     999 P/ FIM "
80 I = I + 1
90  INPUT "XI,YI, XF,YF ";XI(I),Y
     I(I),XF(I),YF(I)
100  IF XI(I) > 999 THEN 190
110  IF YI(I) > 999 THEN 190
120  HPLOT XI(I),YI(I) TO XF(I),Y
     F(I)
130  INPUT "TRACO CORRETO ?  (S/N
     ) ";T$
140  IF T$ = "S" THEN 70
150  HCOLOR= 0
160  HPLOT XI(I),YI(I) TO XF(I),Y
     F(I)
170  HCOLOR= 7
180  GOTO 70
190  HOME :I = I - 1
200  INPUT "ENTRE COM O CENTRO DE
      ROTACAO X0,Y0";X0,Y0
210  HOME
220  INPUT "ENTRE COM O INCREMENT
     O ANGULAR ";K
230  REM
240  REM  CONVERTE PARA RADIANOS
250  REM
260 K = K * 3.1459 / 180
270  INPUT "COLOQUE O NUM. DE FIG
     . ";NF
280 ANG = 0
290  FOR J = 1 TO NF
300 ANG = ANG + K
310  GOSUB 9000
320  FOR M = 1 TO I
330 A = XI(M)
340 B = YI(M)
350  GOSUB 9500
360 X1(M) = X
370 Y1(M) = Y
380 A = XF(M)
390 B = YF(M)
400  GOSUB 9500
410 X2(M) = X
420 Y2(M) = Y
430  NEXT M
440  HCOLOR= 0
450  FOR M = 1 TO I
460  HPLOT X3(M),Y3(M) TO X4(M),Y
     4(M)
470  NEXT M
480  HCOLOR= 7
490  FOR M = 1 TO I
500  HPLOT X1(M),Y1(M) TO X2(M),Y
     2(M)
510  NEXT M
520  FOR M = 1 TO I
530 X3(M) = X1(M)
540 Y3(M) = Y1(M)
550 X4(M) = X2(M)
560 Y4(M) = Y2(M)
570  NEXT M
580  NEXT J
590  END
9000  REM  CALCULO DAS PROJECOES
9010 C(1,1) =  COS (ANG)
9020 C(1,2) =  SIN (ANG)
9030 C(2,1) =  - C(1,2)
9040 C(2,2) = C(1,1)
9050  RETURN
9500  REM  CALCULO DAS COORDENADA
     S
9510 X = X0 + A * C(1,1) + B * C(
     2,1)
9520 Y = Y0 + A * C(1,2) + B * C(
     2,2)
9530  RETURN
```

Para "rodar" esse programa em computadores da família Apple, sugerimos alterar a linha 190 para:

I = I – 1

e suprimir a linha 210.

Uma vez escrito o programa, execute-o e coloque os seguintes pontos: (tecle S ao surgir a mensagem LINHA CERTA? (S/N))

```
10,3,15,3
15,3,25,10
25,10,55,10
55,10,70,15
70,15,10,15
10,15,10,3
40,12,25,20
25,20,15,20
15,20,30,12
30,12,40,12
```

Você formará a mesma figura do número anterior.

Digite agora: 9999,1,1,1 para informar que todos os pontos já foram digitados.

Entre com o centro de rotação. Por exemplo, 140,80.

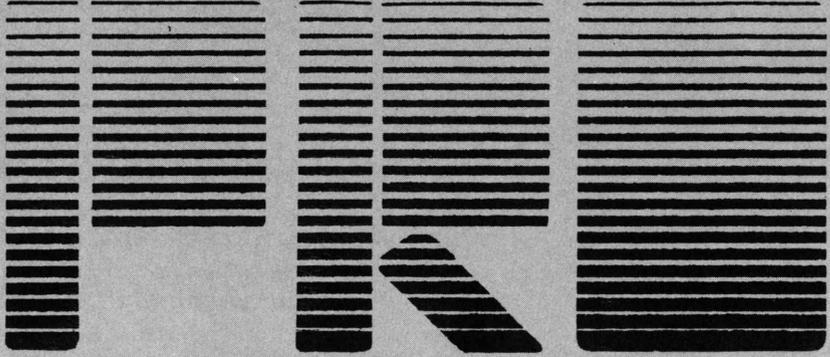
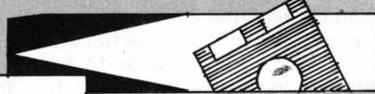
Entre com o incremento angular. Por exemplo, 20.

Entre com o número de figuras. Por exemplo, 18.

Você verá a rotação da sua figura. Para um efeito diferente, modifique a linha 440 para HCOLOR = 7.

Comentários

10 a 60	definição dos vetores de trabalho
70 a 180	colocação da figura
200 a 220	colocação do centro de rotação e incremento angular
240 a 260	conversão para radianos
270	número de figuras
290 a 580	cálculo das coordenadas para a formação das figuras
9000 a 9050	cálculo das projeções para os eixos X e Y
9500 a 9530	cálculo das novas coordenadas

unitron
Ringo
MICRODIGITAL
PHILIPS
CMA
apple
ZIROK
EXATO
Matrix
MICROCRAFT
TEXAS
PRÓ
ELETRÔNICA
SOFTWARE
HARDWARE
SUPRIMENTOS
INSTRUMENTAÇÃO
COMPONENTES
ELETRÔNICOS
Venha conhecer nosso show room e participar ativamente das mais recentes e revolucionárias tecnologias a respeito dos microcomputadores. Além de fazer amigos, eventualmente poderão descobrir o que um Microcomputador poderá fazer por vocês ou pela sua Empresa.
Trata-se de uma perfeita e balanceada estrutura para representar à altura, as principais Empresas de Computadores tais como: Texas, Zirok, Microdigital, Dactari, Polymax, Unitron, Elebra, Aceco, Prológica, Apple II, Savage, Microcraft, CMA, Phillips, Ringo, Matrix...
Intellivision
CERTIFICADO DE REGISTRO NO CADASTRO DE FORNECEDORES DE SUPRIMENTOS DE INFORMÁTICA
SEI
RUA SANTA EFIGÊNIA, 568 - SP - FONE: 221-9055
Kapram PROPAGANDA

O Merge Explicado

parte II

Neste segundo artigo, mostraremos como realmente o computador pode juntar dois programas na sua memória. Além de fornecer uma dica importante, este artigo dá uma série de informações sobre o funcionamento interno do TK 83/85 e seus compatíveis.

Daniel J. R. Nordemamm

O TK 83 ou 85 possui grandes capacidades funcionais. Entre elas, a possibilidade de usar sub-rotina em hexadecimal providencia rapidez e até economia de memória em certos casos. O empacotamento de "string", por exemplo, no 1 REM . . . (endereço fixo do primeiro caractere: 16514) é uma técnica maravilhosa. Porém, quando é preciso manipular volumes importantes de dados surgem outros problemas. Uma solução freqüentemente usada consiste em abaixar RAMTOP (POKE 16389, 126 para criar 512 espaços de memória quando se dispõe da extensão de 16k, por exemplo) e então dispor de um volume bem protegido contra os apagamentos de memória comandados pela instrução NEW, mas evidentemente não é protegido em caso de alimentação desligada, contra a qual não se pode fazer nada. Além disto, o que pode ser uma vantagem vira inconveniente: o conteúdo da memória acima de RAMTOP não é gravado na fita magnética na operação SAVE. Então, para gravar este conteúdo, deve-se transferi-lo abaixo de RAMTOP.

Uma possibilidade que se apresenta é passar o conteúdo da memória acima de RAMTOP para uma tabela, usando LET A(J) = PEEK(B + J) num "loop" conveniente, após o dimensionamento de A( ), sendo B o endereço de RAMTOP. Uma solução melhor e menos dispendiosa é colocar esses "bytes" numa seqüência de caracteres (character string) com LET A$(J) = PEEK (B + J) nas mesmas condições. Ambas soluções são demoradas. O melhor seria transferir os dados, usando POKE (C + J), PEEK(B + J), sendo C e B as respectivas origens dos espaços destinatário e fonte. Ainda melhor e mais rápido seria usar instruções hexadecimais como LD dentro de um "loop", numa sub-rotina escrita em hexadecimal. Para isto é necessário conhecer o endereço do primeiro carácter de uma tabela ou de uma seqüência de caracteres e também levar em conta e saber manejar o fato de este endereço não ser fixo e variar com cada modificação de programa. Este artigo indica como achar este endereço para cada variável (numérica ou de caracteres) e o que fazer para não sofrer as possíveis dificuldades. Além disto, ele indica também como é armazenada cada variável numérica.

```
2——→1    DIM (A$(256) )
         FOR J = 1 TO 256
         LET V = 16513 + J
         LET A$(J) = CHR$ PEEK V
         NEXT J
1——→3    FOR J = 1 TO 256
         LET W = 32000 + J
         POKE W, CODE A$(J)
         NEXT J
3——→1    DIM A(256)
         FOR J = 1 TO 256
         LET W = 32000 + J
         LET A$(J) = CHR$ PEEK W
         NEXT J
2——→3    FOR J = 1 TO 256
         LET V = 16513 + J
         LET W = 32000 + J
         POKE W, PEEK V
         NEXT J
3——→2    FOR J = 1 TO 256
         LET V = 16513 + J
         LET W = 23000 + J$
         POKE V, PEEK W
         NEXT J
```

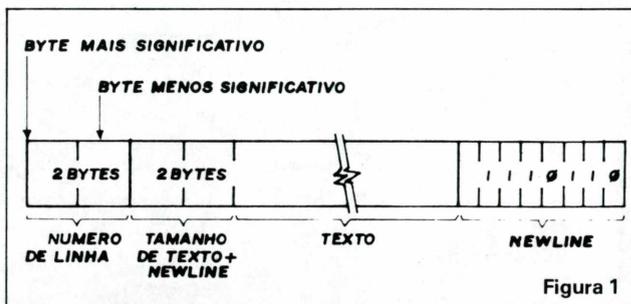
O equivalente dessas instruções pode ser escrito em hexadecimal sem problema, pelo menos com posições fixas (casos 2 e 3). Na configuração da seqüência de caracteres, é necessário procurar o endereço da primeira posição disponível, conforme o que foi descrito no artigo anterior.

Para tratar um conjunto de dados por um programa ao qual ele não está associado, em primeiro lugar carrega-se a memória com os dados, isto é, com os dados e um pequeno programa qualquer que os acompanhe. Transferem-se os dados para posições acima de RAMTOP (para isto é melhor abaixar RAMTOP antes do primeiro carregamento). Em seguida, carrega-se o segundo programa, o programa certo com que os dados serão tratados. Preparam-se as posições para os dados e recuperam-se os dados das posições acima de RAMTOP, ou tratam-se os dados pelo programa, que permanecem acima de RAMTOP, sendo chamados no momento da utilização.

Reunião de Programas

Reunir dois programas (gravados em duas fitas, por exemplo) não é fácil. Toda tentativa de manipular um programa ou algumas linhas de programa como se fossem dados fracassará devido à própria estrutura da linha e das posições dedicadas ao programa.

A estrutura da linha, como indica o manual de instrução, é a que mostramos na figura 1.

Figura 1

A presença de ENTER-NEWLINE e a necessidade da exatidão do tamanho e de ordem dos números das linhas são fatores sensíveis e podem ser causa do não-funcionamento do programa, de impossibilidade de editar (EDIT) uma linha para correção e até o colapso do sistema.

Em primeiro lugar, pdoe-se mencionar que é fácil colocar um programa ou uma seqüência de linhas acima de RAMTOP. O programa seguinte coloca acima de RAMTOP as primeiras linhas até a linha 1000 não-incluída (a condição de elas ocuparem menos de 3*256 "bytes", no caso de RAMTOP = 32000), ou seja:

```
    . . . .
    . . . .
    . . . .
1000 REM AAAA
2000 FOR J=0 TO 768
2010 LET V=16509+J
2020 LET W=32000+J
2030 POKE W,PEEK V
2040 IF PEEK V<>118 OR PEEK (V+1
)<>3 OR PEEK (V+2)<>232 THEN NEX
T J
```

Quando se tem PEEK V = 118, (AND) PEEK (V+1) = 3 e (AND) PEEK (V+2) = 232, sai-se do "loop", pois o número da linha seguinte é 1000 = 3*256 + 232. As linhas que foram transferidas não foram apagadas.

Modificação de linha por POKE

Não é possível passar de maneira simples as linhas de programa que estão acima de RAMTOP para as posições convenientes (entre 16509 e D FILE). Porém, uma pequena "cirurgia", descrita a seguir, é possível nas linhas, embora isto não ajude a transferência procurada. Deve-se tomar uma linha de REM, por exemplo, e cortá-la em duas linhas da REM. Verifique, escrevendo a linha seguinte:

```
1 REM ABCDEFGHIJ
```

que tem a seguinte estrutura:

16509	0	linha 1
16510	1	
16511	12	tamanho 12
16512	0	
16513	234	REM
16514	38	A
16515	39	B
16516	40	C
16517	41	D
16518	42	E
16519	43	F
16520	44	G
16521	45	H
16522	46	I
16523	47	J
16524	118	ENTER

como o pequeno programa apresentado a seguir permite esclarecer:

```
 1 REM ABCDEFGHIJ
10 FOR J=1 TO 16524
20 PRINT J;" ";PEEK J
30 NEXT J
```

A execução da seqüência seguinte

```
 50 POKE 16511,5
 60 POKE 16517,118
 70 POKE 16518,0
 80 POKE 16519,5
 90 POKE 16520,3
100 POKE 16521,0
110 POKE 16522,234
```

transforma a antiga linha 1 em:

```
1 REM ABC
5 REM J
```

ou seja:

16509	0	linha 1
16510	1	
16511	5	tamanho 5
16512	0	
16513	234	REM
16514	38	A
16515	39	B
16516	40	C
16517	118	ENTER
16518	0	linha 5
16519	5	
16520	3	tamanho 3
16521	0	
16522	234	REM
16523	47	J
16524	118	ENTER.

Antes de prosseguir, fica registrada a pequena descoberta tratada neste artigo: pode-se criar nas posições acima do programa e abaixo de D FILE uma região de tamanho constante e de endereços variáveis, quando se aumenta o diminui o volume do programa. Esta região não faz parte do programa e tampouco da parte dedicada à visualização acima de D FILE. O último caractere do programa é o ENTER do fim da última linha e o primeiro da região de visualização é também um ENTER, cujo endereli é D FILE, como se pode verificar através das seguintes instruções:

```
LET D = PEEK 16396 + 256*PEEK 16397
PRINT PEEK (D-1)
PRINT PEEK D
```

que fornecem duas vezes o valor 118.

No meio de um programa qualquer insere-se a linha

```
1000 REM
```

Errata: Merge Explicado

Na Microhobby número 12, por engano, ao invés de publicarmos a primeira lauda do artigo da primeira parte "Mege Explicado", publicamos a primeira lauda da segunda parte. Assim, visando corrigir o erro, publicamos agora o trecho correto. Ele deverá ser substituído no lugar do trecho que começa por "Introdução" e termina em:

```
1 — 2 FOR J = 1 TO 256
      LET V = 16513 + J
      POKE V, CODE A$(J)
      NEXT J
```

Texto Correto:

O TK 83 (ou 85) possui grandes capacidades funcionais. Entre elas, a possibilidade de usar sub-rotinas em hexadecimal providencia rapidez e até economia de memória em certos casos. O empacotamento de "string" (Moreira, 1983), por exemplo, no 1 REM . . . (endereço fixo do primeiro caractere: 16514) é uma técnica maravilhosa. Porém, quando é preciso manipular volumes importantes de dados surgem outros problemas. Uma solução freqüentemente usada consiste em abaixar RAMTOP (POKE 16389, 126 para criar 512 espaços de memória quando se dispõe da extensão de 16K, por exemplo) e então dispor de um volume bem protegido contra os apagamentos de memória comandados pela instrução NEW, mas evidentemente não é protegido em caso de alimentação desligada, contra a qual não se pode fazer nada. Além disto, o que pode ser uma vantagem vira inconveniente: o conteúdo da memória acima de RAMTOP não é gravado na fita magnética na operação SAVE " ". Então, para gravar este conteúdo, deve-se transferi-lo abaixo de RAMTOP.

Uma possibilidade que se apresenta é passar o conteúdo da memória acima de RAMTOP para uma tabela, usando LET A(J) = PEEK(B+J) num "loop" conveniente, após o dimensionamento de A( ), sendo B o endereço de RAMTOP. Uma solução melhor e menos dispendiosa é colocar esses "bytes" numa seqüência de caracteres (character string) com LET A$(J) = PEEK(B+J) nas mesmas condições. Ambas soluções são demoradas. O melhor seria transfe-rir os dados, usando POKE(C+J), PEEK (B+J), sendo C e B as respectivas origens dos espaços destinatário e fonte. Ainda melhor e mais rápido seria usar instruções hexadecimais como LD dentro de um "loop", numa sub-rotina escrita em hexadecimal. Para isto é necessário conhecer o endereço do primeiro caractere de uma tabela ou de uma seqüência de caracteres e também levar em conta e saber manejar o fato de este endereço não ser fixo e variar com cada modificação de programa. Este artigo indica como achar este endereço para cada variável (numérica ou de caracteres) e o que fazer para não sofrer as possíveis dificuldades. Além disto, ele indica também como é armazenada cada variável numérica.

que tem a seguinte estrutura:

J − 1	118	ENTER
J	3	linha 1000
J + 1	232	
J + 2	2	tamanho 2
J + 3	0	
J + 4	234	REM
J + 5	118	ENTER.

Coloca-se um ENTER na posição J, pela execução das seguintes linhas:

```
10 FOR J=116509 TO 32000
20 IF PEEK (J-1)<>118 OR PEEK
J<>3 OR PEEK (J+1)<>232 THEN NEX
T J
30 POKE J,118
40 STOP
```

Fez-se desaparecer da listagem toda a parte do programa acima da linha 1000 inclusive. Não é mais possível, por enquanto, executar esta parte de programa. Mas ela não foi perdida, pois ficou numa região protegida, a qual foi criada entre o endereço J e D FILE. O conteúdo desta região protegida e o seu comprimento ficaram inalterados mesmo quando se encurtou ou aumentou o programa visível na listagem, embora, agora, D FILE e J tenham variado. A volta à listagem do programa desaparecido não apresenta dificuldade, mesmo após modificações no programa visível; basta executar as linhas:

```
10 FOR J=16509 TO 18000
20 IF PEEK (J-1)<>118 OR PEEK
J<>118 THEN NEXT J
30 POKE J,3
```

Coloca-se na zona protegida de 118 118 até D FILE um programa que havia sido reservado, nas posições acima de RAMTOP. Uma condição para evitar complicações consiste em fazer com que o número da primeira linha deste programa seja superior ao número da última linha do programa já residente, que continuará sendo no exemplo dado:

```
990 REM
```

A região reservada, acima de 118 118, deve ter um tamanho suficiente para comportar o programa a ser inserido, o qual, armazenado nas posições acima de RAMTOP, começa por uma linha 1000 qualquer e termina, por exemplo, pela linha:

```
2048 REM
```

Em seguida, transfere-se o programa das posições acima de RAMTOP para a zona reservada, através da seguinte seqüência:

```
200 FOR J=16509 TO 18000
210 IF PEEK (J-1)<>118 OR PEEK
J<>118 THEN NEXT J
250 FOR K=0 TO 768
260 LET W=32000+K
270 LET V=J+K
275 POKE V,PEEK W
280 IF PEEK W<>118 OR PEEK (W+1
)<>8 OR PEEK (W+2)<>0 THEN NEXT
K
```

Finalmente, realiza-se o procedimento necessário para a seqüência acrescentada ao programa tenha o seu último ENTER seguido de mais um ENTER, marcando assim o fim do programa completo:

```
290 POKE V+1,118
```

Na verdade, não era necessário ter criado a zona reservada acima de 118 118 para poder baixar parte do programa, a partir das posições acima de RAMTOP, como se verá mais adiante.

Transferências simples de programas entre RAMTOP e 16509

Para a reunião de programas diferentes em um só, a operação fundamental é a transferência de um programa de 16509 . . . até RAMTOP, o que deve ser realizado sem perda do conteúdo da memória. Apresentam-se, a seguir, três programas que realizam tais transferências de uma maneira relativamente simples. O primeiro serve para realizar transferências nos dois sentidos: de baixo para cima (de 16509 . . . para RAMTOP . . .) e de cima para baixo DOWN (de RAMTOP . . . para 16509 . . .), usando os valores 1 e 0, respectivamente; para o parâmetro V.

```
   1 REM TRANSFERENCIA DE PROGRA
MAS ENTRE 16509 E RAMTOP
  99 REM
 100 FAST
 110 PRINT "UP:1,DOWN:0"
 120 INPUT U
 130 POKE 16389,125
 140 FOR K=16509 TO 16509+(U=0)*
768
 150 IF PEEK K<>118 OR PEEK (K+1
)<>4 THEN NEXT K
 160 LET V=16509*U+125*256*(U=0)
 170 LET W=125*256*U+(K+1)*U=0
 180 FOR J=0 TO 678
 182 LET X=V+J
 184 LET Y=W+J
 190 POKE Y,PEEK X
 195 IF U=0 AND PEEK (Y-1)=118 T
HEN POKE Y,PEEK Y+4
 200 IF PEEK X<>118 OR PEEK (X-4
)<>99 THEN NEXT J
 210 POKE Y+1,118
 220 STOP
1024 REM AAAAAAAAAAAAAAAAAAAA.
```

Levando em conta o que já foi explicado, este programa precisa de poucos esclarecimentos. A variável U pode substituir a função lógica (U = 1), uma vez que U só toma os valores 1 e 0. A linha 195 foi acrescentada ao programa para adicionar 1024 (acrescentando 4 ao "byte" mais significativo) aos números de linha no momento da descida, a fim de evitar inversões a repetições na seqüência dos números de linha. Evidentemente, o volume de programa que segue a linha 1024 deve ser maior do que o programa a ser transferido neste lugar.

```
 768 REM
 900 FAST
 930 POKE 16389,125
 980 FOR J=0 TO 3*256
 985 LET X=16509+J
 990 POKE 32000+J,PEEK X
1000 IF PEEK X<>118 OR PEEK (X-5
)<>3 OR PEEK (X-4)<>0 THEN NEXT
J
1010 PRINT J
1020 STOP
```

O segundo programa, mais simples ainda, realiza a transferência de 16509 para RAMTOP.

```
 115 LIST 1000
 900 FAST
 940 FOR J=16509 TO 18000
 945 IF PEEK (Y-1)=118 THEN POKE
 Y,PEEK Y+4
 950 IF PEEK K<>118 OR PEEK (K+1
)<>4 OR PEEK (K+2)<>0 THEN NEXT
K
 970 LET W=K+1
 980 FOR J=1 TO 768
 982 LET X=V+J
 984 LET Y=W+J
 990 POKE Y,PEEK X
1000 IF PEEK X<>118 OR PEEK (X-5
)<>3 OR PEEK (X-4)<>0 THEN NEXT
J
1010 POKE Y+1,118
1024 REM 00000
```

O terceiro programa realiza a operação inversa, transferindo de RAMTOP para 1024 REM.

Apresenta-se, a seguir, um processo prático para incorporar um programa a outro. Os dois programas a integrar, A e B, estão gravados em fitas diferentes. A seqüência das operações é a seguinte:

1) Abaixar RAMTOP para reservar um espaço suficiente onde caiba B. Por exemplo, POKE 16389, 125 reserva 3 vezes 256 "bytes", o que é em geral suficiente para alojar um pequeno programa de demonstração. Segue-se com NEW, ENTER.
2) Carregar o programa B utilizando a instrução LOAD "B" ou equivalente.
3) Acrescentar a B algumas linhas cuja execução transferirá B para posições acima de RAMTOP (essas linhas não devem ser transferidas).
4) Carregar o programa A utilizando a instrução LOAD "A" ou equivalente.
5) Acrescentar a A algumas linhas cuja execução transferirá B para a região de programa. Aos números de linha de B deve ser acrescentado um valor, múltiplo de 256, que garanta que o número de sua primeira linha seja maior do que o da última linha de A (isto pode ser feito somando um inteiro ao primeiro "byte" do número de linha). Respeita-se também a regra que impõe a existência no fim da última linha de B para indicar o fim do programa ativo.

Conclusão

Com esta metodologia de transferência e de reunião de dados e de programas, os leitores interessados poderão utilizar com mais eficiência a memória dos seus TK 83, 85 e compatíveis. Foram indicados os procedimentos que parecem mais seguros, mas isto não dispensa os devidos cuidados, pois um pequeno erro ou uma invasão imprevista na área de programa ativo podem causar danos ao conteúdo da memória. Boa sorte.

Daniel J. R. Nordemann é pesquisador do Instituto de Pesquisas Espaciais

Se os microcomputadores tem tomado quase que exclusivamente as páginas das publicações especializadas em informática, nós acreditamos, por outro lado, que as calculadoras são, ainda, uma ferramenta muito importante, principalmente para o estudante e o profissional liberal. Sob este ponto de vista, estamos abrindo mais esta seção, que pretende apresentar, entre outros, os seguintes assuntos:

— estruturas e detalhes técnicos de calculadoras;

— técnicas de programação básica e avançada;

— aplicativos técnicos, científicos e comerciais;

— programas em geral;

— livros sobre calculadoras (comentários).

Abrimos a seção com três artigos: um apresentando a HP-41 em breve discussão, outro introduzindo a notação polonesa reversa e, finalmente um terceiro tópico abrangendo a determinação de raizes de equações d eum modo clássico, porém muito eficiente.

Calculadoras

Wilson José Tucci
José Eduardo Moreira
Colaborações de
Luiz Carlos Szente
Daniel R. Falconer

Apresentando a HP-41

Quando a Hewlett-Packard introduziu sua calculadora HP-41C há alguns anos atrás, ela não estava apenas criando mais uma calculadora, ela estava introduzindo toda uma nova filosofia em programação pessoal. Começavam a aparecer os primeiros computadores portáteis, e a Hewlett-Packard fazia sucesso com suas avançadas calculadoras HP-34 e HP-67. Lançou, então, a HP-41C, com finalidade de ser uma calculadora de alta qualidade, e ao contrário das calculadoras existentes, possuir grande flexibilidade e capacidade de expansão. Foi também uma das primeiras calculadoras portáteis a ter capacidade de manipular dados alfanuméricos.

A flexibilidade e capacidade de expansão de HP-41 fica evidenciada pelas seguintes características:

1) capacidade de expansão de memória
2) disponibilidade de vários módulos que podem:
a) conter programas aplicativos de diversas áreas
b) adicionar mais funções ao seu repertório
c) permitir tratamento de arquivos e de texto
3) grande números de periféricos, entre os quais se destacam:
a) leitora de cartões magnéticos
b) impressora térmica
c) interface de vídeo
d) cassete digital, para armazenar programas e arquivos
e) HPIL, uma interface para comunicações em geral

Claro que toda essa flexibilidade da HP-41 tinha que ter um preço. Para conseguir tudo isso da sua calculadora a Hewlett-Packard optou por um projeto de software intensivo, ou seja, um hardware relativamente simples, controlado por uma grande quantidade de software. Se isso permite freqüentes expansões da máquina, simplesmente incluindo mais software, resultou numa calculadora mais lenta do que a tecnologia da época permitia. A freqüência do clock da HP-41 é de 350kH, e do Apple é 1023 kHz. Além disso, num loop vazio (em BASIC) o Apple é 100 vezes mais rápido do que a HP-41. Mesmo assim, a velocidade da HP-41 é satisfatória para a maioria das aplicações que se espera de uma calculadora portátil.

Um outro ponto a favor do HP-41 e das calculadoras Hewlett-Packard, é a qualidade do seu hardware e a segurança da garantia, mas, como a HP-41 é baseada em muito software, é comum problemas em que a HP-41 se perde ou faz besteiras que travam a calculadora, principalmente quando se faz programações avançadas.

A família HP-41 é composta por 3 calculadoras: a HP-41C, a HP-41CV (a única vendida no Brasil, no momento) e a HP-41CX (a mais avançada, ainda não vendida no Brasil). A diferença entre essas calculadoras é mais uma questão de grau do que de gênero: a HP-41CV tem, como única diferença visível 5 vezes mais memória que a HP-41C, já a HP-41CX traz embutida uma série de funções que requerem módulos extras para serem implementadas nos outros dois modelos.

Nos próximos números, apresentaremos aos leitores, mais detalhes de fabricação, *microcodes,* display, funções, etc., dessa notável macrocalculadora.

Técnicas de Programação para HP-41

Aula 1 — Notação polonesa reversa.

A primeira característica "estranha" que salta aos olhos de alguém que está usando pela primeira vez uma calculadora HP, é o modo não usual de entrada de dados e operações, baseado numa notação matemática chamada RPN (Reserve Polish Notation). No começo a RPN parece apenas uma complicação desnecessária, contudo as pessoas acabam tão acostumadas com sua facilidade e eficiência que quando tem que lidar com calculadoras usuais, têm dificuldades.

Sistemas de notação.

1. Notação infixa: geralmente escrevemos expressões matemáticas na forma:

operando operador operando

exemplos:

a + b
(a*b) + c
(a + b) * (c + d)
(a + b*c) — (d/f + e)

Essa forma de notação chama-se notação infixa, e como você deve ter notado, é preciso adotar convenções de precedência e/ou uso de parênteses para efetuar corretamente a expressão.

Essa necessidade de parênteses e precedência complica, de um modo geral, a avaliação de expressões em computadores (grandes e pequenos). Uma outra forma mais eficiente de se escrever expressões matemáticas, é a notação polonesa, que pode aparecer sob duas formas:

a) operador (operando) operando (notação polonesa)
(operando operando operador) (notação polonesa reversa)

A seguinte tabela mostra três notações equivalentes para quatro expressões:

infixa

a + b
(a*b) + c
(a + b) * (c + d)
(a + b*c) — d/f + e)

polonesa

+ ab
+ * abc
* + ab + cd
— + a*bc + /dfe

polonesa reversa

ab +
ab*c +
ab + cd + *
abc* + df/e + —

Note que, nas duas notações polonesas, a ordem das operações está perfeitamente determinada, sem necessidade de parênteses e regras de precedência, sendo a polonesa reversa (RPN) melhor para se introduzir dados.

A notação polonesa foi apenas uma das inúmeras contribuições ao conhecimento humano feita pelo fantástico matemático polones Jan Łukesiewicz, nascido em 21 de dezembro de 1878 na cidade de Lwow, Polonia. Se formou em matemática e filosofia na Universidade de Lwow, onde lecionou até 1915. Por volta de 1910, ele desenvolveu sua "notação polonesa". Entre 1915 e 1944, trabalhou na Universidade de Varsóvia, período em que publicou cêrca de 80 trabalhos de psicologia, matemática e filosofia (principalmente lógica). Se esconde deu na cidade alemã de Münster entre 1944 e abril de 1945, quando os americanos libertaram a cidade. Mudou-se então para a Bélgica, onde morou até 1946, quando então foi convidado para a cadeira de Lógica na Academia Real Irlandesa e fazer parte do Instituto de Estudos Avançados, ambos em Dublin, Irlanda. Lá publicou mais dez trabalhos até sua morte em 13 de fevereiro de 1956.

Além de desenvolver a notação polonesa, Jan Łukesiewicz também desenvolveu o cálculo proporcional de três valores (verdadeiro, falso, não sei), que é um método para calcular o valor de uma sentença lógica. As "tabelas-verdades", comuns nos cursos de Lógica são um subproduto desse cálculo profissional.

Łukesiewicz viveu o bastante para ver o começo da era do computador e implementação de algumas de suas idéias (ainda hoje fundamentais para a ciência do computador).

Na próxima aula, veremos como a notação polonesa é implementada e usada na HP.

BIBLIOGRAFIA

THE HEWLETT PACKARD Personal Calculador Digest. Volume 7.
TREMBLAY, Jean Paul: *Introduction to Computer Science*
THE HEWLETT Packard Personal Calculator Digest. Volume 7.

Busca de raizes para HP-41 (Método de Newton)

Sendo a HP-41 uma calculadora predominantemente científica, é justo que nosso primeiro artigo sobre ela trate de um assunto científico: determinação das raizes de uma equação.

Existem vários métodos para se calcular as raizes de uma equação. Um método bastante rápido é o método de Newton (ele mesmo, o da maçã). O método se baseia no seguinte:

se conhecermos um primeiro chute da raiz, x_1, podemos calcular um chute melhor, x_2, pela fórmula:

$$x_2 = x_1 - \frac{fx_1)}{f'(x_1)}$$

ou, de forma mais geral:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

onde:

$$f'(x_n) = \frac{(f(x_n + dx) - f(x_n)}{dx}$$

onde dx é um número bem pequeno.

O algoritmo para tal cálculo pode ser assim escrito:

1. inicialize ITMAX (o número máximo de interações) e X (chute) e EPS (precisão desejada)
2. repita os passos 3. a 5. para I = 1,2, 3,. . ., ITMAX
   3. RAIZ X − F(X)/f'(X)
   4. se f (RAIZ) EPS vá para o passo I
   5. X RAIZ
6. raiz não convergiu em tempo
7. fim
8. mostre a raiz
9. fim

Implementação do algoritmo na HP-41:

1) O usuário terá que escrever um programa que calcule a função desejada. Tal programa deve calcular f(x), onde x é o conteúdo do registrador X, f(x) deve ser colocado no registrador X. Exemplo, para calcular a raiz da função x.ln(n + 1) devemos fazer, por exemplo, o seguinte programa:

```
01♦LBL "FF"
02 STO 08
03 1
04 +
05 LN
06 RCL 08
07 *
08 END
```

Quando o programa para calcular as raízes perguntar qual a função, o usuário deve responder com o nome que deu ao programa dele. O programa do usuário não deve usar os registradores 00 a 07.

2) Usaremos os seguintes registradores para representar as respectivas variáveis:

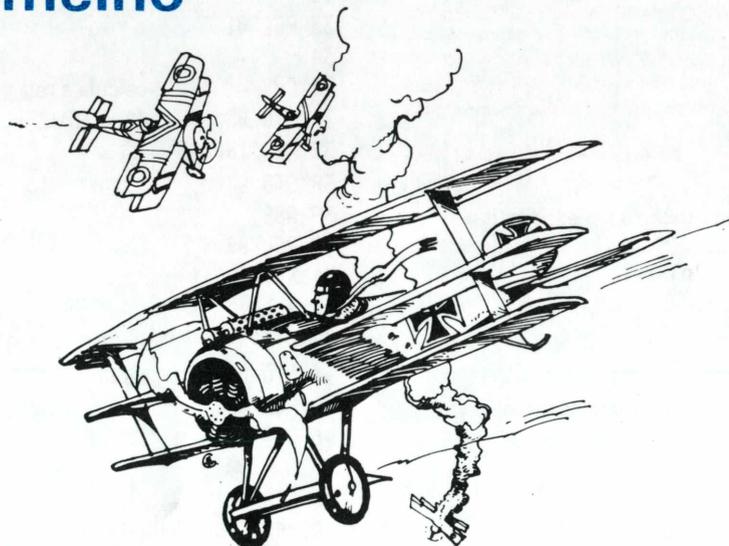
00 = ITMAX
01 = X
02 = RAIZ
03 = PRECISÃO
04 = f(x)
05 = nome da função
06 = dx (valor pode ser redefinido conforme o usuário necessitar)

O programa:

```
01♦LBL "NEWTON"
02 1 E-6
03 STO 06          ; 06 = dx
04 "FUNCAO ?"
05 AON
06 PROMPT
07 AOFF
08 ASTO 05         ; nome da função a ser executada
09 "ITERACOES ?"
10 PROMPT
11 STO 00          ; número máximo de interações
12 "PRECISAO ?"
13 PROMPT
14 STO 03          ; quão perto de zero desejamos
15 "CHUTE ?"
16 PROMPT
17 STO 01          ; 01 = X
18♦LBL 00          ; começo do laco
19 RCL 01
20 XEQ IND 05
21 STO 04          ; 04 = f(x)
22 RCL 01
23 RCL 06
24 +
25 XEQ IND 05      ; coloca f(x + dx) no registrador X
26 RCL 04
27 -
28 RCL 06          ; coloca f'(x) no registrador X
29 /
30 RCL 04
31 X<>Y
32 /
33 RCL 01
34 -
35 CHS             ; calcula a raiz
36 STO 02          ; 02 = RAIZ
37 XEQ IND 05
38 STO 07
39 ABS
40 RCL 03
41 X<>Y
42 -
43 X>0?            ; se EPS    f(RAIZ)  vá para apresentação
44 GTO 01          ; repita
45 RCL 02
46 STO 01
47 DSE 00
48 GTO 00
49 "RAIZ NAO CONVER"
50 "FGIU"
51 AVIEW
52 PSE             ; mensagem que só aparece se não convergiu
53♦LBL 01
54 "RAIZ="
55 ARCL 02
56 AVIEW
57 STOP
58 "FRAIZ="        ; mensagem mostrando a RAIZ e f(RAIZ)
59 ARCL 07
60 AVIEW
61 END
```

Exemplo do uso do programa para calcular a raiz de $x^2 - 4$.

Passo	Instrução	Dados	Visor
1	introduza o programa que calcula a função		01♦LBL "FF" 02 X↑2 03 4 04 - 05 END
2	inicialização	XEQT NEWTON	
		nome do programa	? FUNÇÃO FF R/S
		máximo de iterações	?ITERAÇÕES 10 R/S
		precisão desejada	?PRECISÃO 0.00001 R/S
		chute inicial	?CHUTE 10
3	Calcule a raiz		R/S
4	Resultado		RAIZ = 2.00 R/S FRAIZ = 0.00

A volta do Barão Vermelho

Gustavo Egídio de Almeida

Durante a primeira Guerra Mundial, existiram muitos heróis, que se destacaram pelos seus atos de bravura. A aviação teve também seus heróis, mas por mais que se force a memória, só se consegue lembrar de dois deles: Otto von Richthofen — o Barão Vermelho — e seu inimigo, o Comandante Brown.

Muito do que se sabe sobre suas aventuras está misturado a lendas, que tendem a mistificá-los. Entretanto, só merece uma lenda quem de alguma forma correspondeu aos fatos por ela narrados.

No caso destes dois, todos os fatos narrados — verídicos ou lendários — apontam para três características: habilidades no manejo dos aviões, coragem e nobreza.

A aviação, nesta época, estava apenas engatinhando. Pilotar já era um ato de coragem, dada a precariedade dos aviões. Os raides aéreos eram verdadeiras batalhas entre pilotos, onde o que importava era a habilidade de cada um. Dentre estes pilotos, alguns se destacavam e recebiam a denominação de Ás.

Devido às características peculiares dos combates de ataques aéreos, bem diferentes do combate em terra, o confronto entre dois ases assumia a característica de um "justa" medieval, com um código de ética baseado no respeito mútuo (se bem que algumas vezes ele era esquecido).

Durante um dos raides, o comandante da esquadrilha alemã, antes da partida, recomendou a seus pilotos que camuflassem seus aviões. O Barão, além de desobedecer esta ordem, pintou seu avião de vermelho, fazendo o oposto do que lhe tinham pedido. A partir de então, o Barão Otto von Richthofen passou a ser conhecido como "o Barão Vermelho", figura respeitada e temida pelos pilotos ingleses.

Entre seus inimigos encontrava-se o comandante Brown, um piloto tão hábil quanto ele e, em cada raide aéreo, havia um desafio mútuo, uma rivalidade maior que o simples fato dos dois estarem lutando em lados contrários da guerra. Isto durou muito tempo, até que o comandante Brown, em um dos raides, finalmente o derrotou, derrubando seu avião.

Alguns dias após a morte do Barão, um avião inglês jogou no campo de pouso alemão uma coroa de flores com uma medalha inglesa por bravura, destinada ao Barão Otto von Richthofen, reconhecendo em sua pessoa um herói digno desta distinção, não importando ser ele inimigo.

A volta do Barão Vermelho

Você deve estar lembrado, no número 0 da *Microhobby* tinha como programa central, que deu origem ao nosso personagem-símbolo, o *Barão Vermelho.*

Apesar de ser um programa simples, contendo apenas a linguagem BASIC do TK 83, despertou bastante interesse por parte dos leitores, principalmente aqueles que naquela época estavam entrando no mundo da computação e mal conheciam a linguagem Basic.

Como o adotamos como personagem-símbolo, voltamos com uma nova versão do famoso *Barão Vermelho.*

Esta versão é diferente da primeira em quase todos os aspectos. Vamos citar as principais:

— na primeira versão, como deve estar lembrado, você era o capitão Brown, pilogando um Sopwith Camel e agia contra o famoso Barão Von Richthofen, o *Barão Vermelho;*

— agora as coisas se inverteram e você pilota o famoso triplano Fokker DR-1 e trava uma verdadeira batalha contra o capitão Brown e seu biplano Sopwhith Camel;

— o visual do jogo é apresentado em três dimensões. Você está dentro da cabine de seu Fokker DR-1, tendo à mira de sua metralhadora bem a sua frente. Sua munição é limitada e composta de 30 tiros no seu total. Ao esgotar esta munição, o jogo se encerra. (fig. 1)

Seu inimigo está sempre à sua frente e você tem que enquadrá-lo em sua mira para poder abatê-lo.

Fig. 1

Para mover seu triplano, use as setas contidas nas teclas e para atirar, pressione a tecla 0 (TK 83/85).

O programa para o TK 83 (com expansão) ou 85

Este programa apresenta três subrotinas em Linguagem de Máquina, que servem para agilizar e dar um efeito gráfico mais interessante ao jogo. A primeira rotina reside na linha 1 REM e serve para memorizar uma tela inteira e guardá-la em algum endereço da RAM (é chamada com o comando RAND USR 16514).

Quando a linha 2 REM é requisitada através do comando RAND USR 16534, a tela memorizada na rotina anterior (ou seja, na 1 REM), é colocada instantaneamente na tela economizando um grande espaço de tempo durante o jogo.

A rotina existente na linha 3 REM serve para inverter os caracteres da tela num determinado número de vezes.

Para entrar com as 3 linhas REM, siga as seguintes instruções:

— as linhas 1, 2, 3 REM devem conter cada uma o número de caracteres 14, 13 e 38, respectivamente.

— digite agora o monitor assembler da figura 2 e rode o programa com o comando RUN (NEW LINE). Entre com os seguintes códigos em Linguagem de Máquina (para parar, digite P):

16514	33	16559	12
16515	48	16560	64
16516	117	16561	35
16517	237	16562	6
16518	91	16563	5
16519	12	16564	197
16520	64	16565	6
16521	19	16566	145
16522	1	16567	126
16523	214	16568	198
16524	2	16569	0
16525	237	16570	254
16526	176	16571	118
16527	201	16572	40
16534	42	16573	3
16535	12	16574	198
16536	64	16575	128
16537	17	16576	119
16538	48	16577	35
16539	117	16578	16
16540	1	16579	243
16541	214	16580	193
16542	2	16581	16
16543	35	16582	237
16544	237	16583	58
16545	176	16584	32
16546	201	16585	78
16553	62	16586	51
16554	8	16587	71
16555	50	16588	16
16556	32	16589	221
16557	78	16590	201
16558	42		

```
   1 REM ................
   2 REM ...............
   3 REM .......................
...........
  10 LET F=16514
  20 SCROLL
  25 PRINT F,
  30 INPUT A$
  35 IF A$="P" THEN STOP
  40 LET A=VAL A$
  50 POKE F,A
  55 PRINT PEEK F
  60 IF PEEK F=201 THEN LET F=F+
6
. 65 LET F=F+1
  70 GOTO 20
```

Fig. 2

— apague as linhas de 10 a 70 e digite a listagem em BASIC apresentada a seguir. Ao digitar os caracteres gráficos contidos na listagem do programa, preste bastante atenção. Para gravar o programa dê o comando RUN 370 seguido de NEW LINE. O programa é auto explicativo. *Boa sorte Barão Von Richthofen!...*

```
   1 REM 5K? GOSUB ?£RND(▀CHR$ ▀
 GOSUB ▓TAN
   2 REM E£RND)K?▀CHR$ ▀7 GOSUB
▓TAN
   3 REM Y▒M4?E£RND7▄█VAL ▄█▀LEN
 █?7( NEXT AT ( GOSUB U4?X?(<>TA
N (
   4 LET SCR=16514
   5 LET CPY=16534
   7 POKE 16418,0
  10 FOR F=0 TO 4
  20 PRINT TAB F*2;"▀▄";TAB 30-F
*2;"▄▀"
  30 NEXT F
  40 PRINT TAB 10;"▛▀▀▀▀▀▀▀▀▀▀▜"
  50 FOR F=0 TO 5
  60 PRINT TAB 9-F;"▌";TAB 22+F;
"▌"
  70 PRINT TAB 9-F;"▌";TAB 22+F;
" ▌"
  80 NEXT F
  90 FOR F=0 TO 3
 100 PRINT "████████████████████
███████████"
 110 NEXT F
 120 PRINT AT 15,14;"▄█▄";AT 16,
15;"▌";AT 17,13;"▄███▄"
 130 LET L=USR CPY
 140 LET S=10
 150 LET K=0
 160 LET X=INT (RND*32)
 170 LET Y=INT (RND*18)
 180 LET P=USR SCR
 190 PRINT AT Y-1,X;"▄▀";AT Y,X;
"▄█"
 195 IF S=0 THEN GOTO 500
 200 IF INKEY$="0" THEN GOTO 300
 210 LET X=X+INT (RND*3)-1+(INKE
Y$="5")-(INKEY$="8")
 220 LET X=X+2*((X<0)-(X>30))
 230 LET Y=Y+INT (RND*3)-1+(INKE
Y$="7")-(INKEY$="6")
 240 LET Y=Y+2*((Y<0)-(Y>17))
 250 GOTO 180
 300 PRINT AT 15,14;"***"
 310 LET S=S-1
 320 IF X<17 AND X>13 AND Y=15 T
HEN GOTO 340
 330 GOTO 210
 340 LET K=K+1
 345 LET L=USR 16553
 350 POKE 30643,K+156
 360 GOTO 180
 370 SAVE "B/R"
 380 PRINT TAB 7;"BARAO VERMELHO
 ▓▓"
 390 PRINT AT 10,0;"PARA CONTROL
AR SEU TRIPLANO USE AS TECLAS 5,
6,7,8.                PARA DESTRUI
R SEU INIMIGO USE A TECLA 0."
 400 PRINT AT 21,0;"TECLE NEW LI
NE PARA INICIAR."
 410 INPUT I$
 415 PRINT AT 0,0;
 417 CLS
 420 RUN
 500 PRINT AT 10,7;"SUA MUNICAO
ACABOU"
 600 PRINT AT 23,4;"TECLE P PARA
 OUTRO JOGO"
 700 IF INKEY$<>"P" THEN GOTO 70
0
 750 CLS
 800 RUN
```

Computação Gráfica

Dando início a uma nova fase da Microhobby, decidimos fazer esta edição com um tema bastante interessante: computação gráfica. Para isto, preocupamo-nos em transmitir aos leitores informações sobre o que é computação gráfica, sistemas de grande porte e suas aplicações, recursos de microcomputadores e seus usuários. Com este intuito, procuramos pesquisar tudo o que já foi feito sobre o assunto, entrevistar especialistas na área e contactar as universidades. Assim sendo, chegamos a esta matéria (abrangente) que esperamos atenda àqueles objetivos iniciais de sua realização, assim como elucide a curiosidade vocês.

Ana Lúcia de Alcântara

Traçar gráficos, fazer projetos e criar obras artísticas são tarefas inerentes a determinadas categorias profissionais, que a partir da evolução tecnológica dos computadores, tornaram-se menos desgastantes. A utilização do computador em áreas que requerem cálculos complexos e exigem diversas formas de representação gráfica ou de variação de imagens, possibilitou uma diminuição da margem de erro nos resultados, tornando-os mais eficazes em sua configuração final.

A perfeita combinação entre hardware, software e periféricos auxiliares fornecem ao usuário dos sistemas de computação gráfica, inúmeros recursos que lhes permitem efeitos de qualidade superiores àqueles obtidos nos seus métodos tradicionais (manuais) de trabalho.

Campos de aplicação

A computação gráfica alcançará, segundo estimativa de Maxine D. Brown, consultora independente nos Estados Unidos (especialista em tecnologia de comunicação por computadores gráficos), um índice de crescimento para esta década em torno de 40 a 65 por cento e já oferece inúmeras opções em termos de hardware, indo desde os micros (como os da família Apple, IBM PC, TRS-Color e outros), embora com algumas limitações, como baixa disponibilidade de software, capacidade de memória, etc., até os minicomputadores como, por exemplo, os modelos mais atuais do VAX-750 que já possuem capacidade de gerar cerca de 16 milhões de tonalidades diferentes de cores, quando programados para esta finalidade e com periféricos apropriados.

A computação gráfica possue três grandes campos de aplicações: através do sistema CAD (Computer Aided Design — projetos auxiliares por computador) em projetos de equipamentos mecânicos; na área eletro-eletrônica; em desenho de equipamento de controle e comando; na elaboração de gráficos científicos; em arquitetura, entre outros. Este sistema permite trabalhar tanto com dados bidimensionais (estruturas planas — mapas e na área de eletrônica, em circuitos integrados) como com dados tridimensionais (topografia, arquitetura e mecânica), porém, neste caso, surgem alguns problemas de visibilidade e superposição de objetos.

Outra aplicação dos sistemas gráficos é no processamento de imagens, que fornece recursos para a identificação de contornos e extremidades da imagem, além de permitir o realce de partes em movimento. Um exemplo de processamento de imagens é a aplicação na área militar, no reconhecimento do terreno; é também utilizada na captação de imagens transmitidas por sinais de satélites. O processamento de imagens encontra todavia, algumas restrições quanto a entrada de dados, pois necessita de uma base de dados maior que os outros sistemas já que, por exemplo, para se gerar uma tela multicolorida pode ser necessário até 1 Mbyte de memória para definí-la.

Em aplicações que visam o campo das artes e do entreterimento, os sistemas gráficos têm atuado como meio de expressão de artistas e também na produção de desenhos animados, logotipos e produção de filmes comerciais ou de longa metragem (exemplos TRON, Guerra nas Estrelas, entre outros). As técnicas de animação usadas nestes tipos de aplicações são também divididas em duas classes: bidimensionais, onde os desenhos são feitos manualmente e depois as imagens são interpoladas pelo computador, e tridimensionais, envolvendo um trabalho de manipulação de bonecos e modelos pré-definidos. Este processo foi usado em diversas produções de filmes de ficção científica como por exemplo, 2001 — Uma Odisséia no Espaço e o Cristal Encantado.

Problemas

Porém, a maioria destes sistemas possuem problemas tanto na introdução como saída e manipulação dos dados transmitidos ao computador. Para superar estes problemas, os fabricantes de sistemas gráficos pesquisaram e chegaram aos chamados dispositivos de entrada e saída que auxiliam os usuários na elaboração das imagens digitais. Para entrada de dados existem no mercado americano e agora, (embora timidamente, no brasileiro) dispositivos como as mesas digitadoras — indicadas para a digitalização e entrada de dados —, o joystick e o mouse (que já são utilizados em microcomputadores) que servem para a movimentação do cursor na tela e o light-pen, que possibilita ao usuário apontar o desenho diretamente na tela. Para o auxílio na saída de dados, os dispositivos são separados em duas categorias, os ativos (que permitem uma interação maior entre o programador e o computador) e os passivos como impressoras, plotters (os mais utilizados, devido ao seu princípio básico nas coordenadas x e y auxiliam no traçamento digital da figura) e os displays.

Esquema de trabalho numa estação gráfica

O processamento gráfico de um grande sistema de computadores como o VAX-750 é todo ele realizado numa "workstation" ou estação gráfica. Nela, estão incluídos o dispositivo de visualização assim como os demais (entrada e saída) e todos estão acoplados aos "host", que controla todo o processamento central. A inteligência destas "workstations" desenvolveu-se a tal ponto, que hoje em dia quase não há condições de definir-se os limites precisos entre estas e o "host", pois a imposição destes limites levaria à uma redução na eficiência dos sistemas.

A maioria dos sistèmas gráficos são interativos em função da necessidade de comunicação existente entre operador e o sistema. Esta necessidade surge porque a comunicação por meio de ima-

Exemplo de interação entre o sistema e o operador numa estação gráfica.

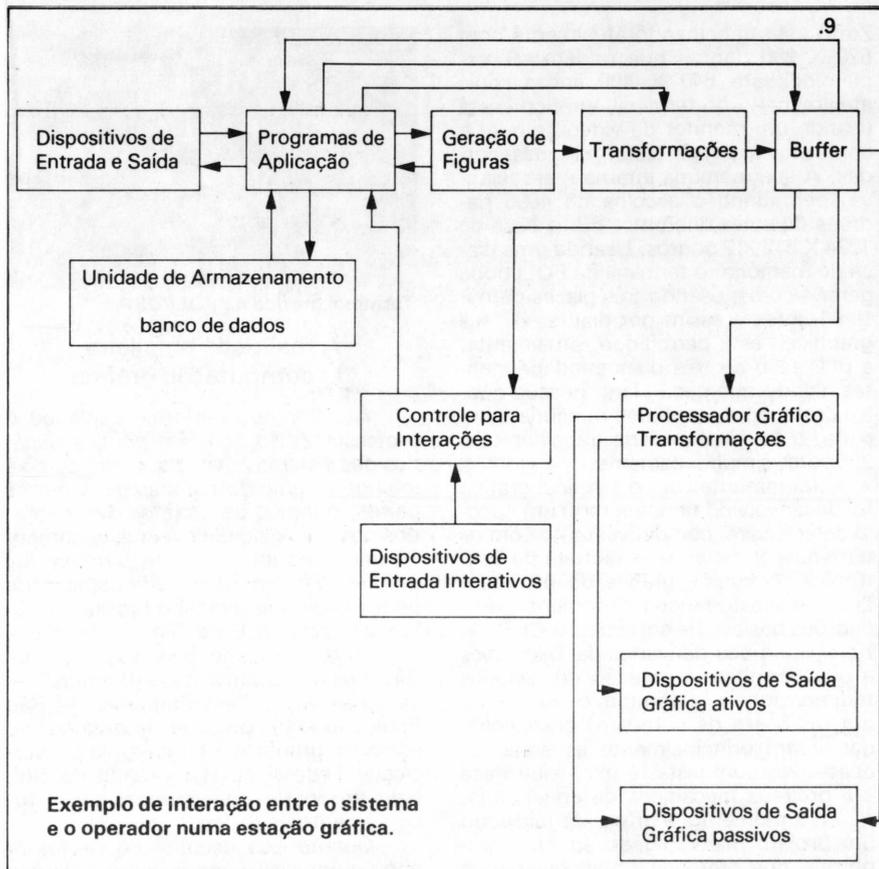



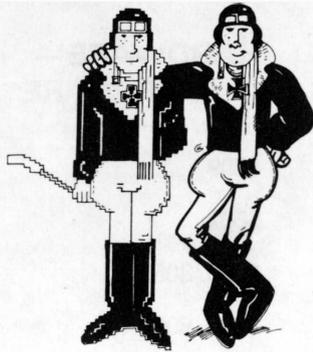
Figura 1

gens oferece uma maior rapidez e sua quantidade de informações também é muito superior a de um processamento convencional. Um exemplo de interação entre o operador e os sistemas gráficos (conforme o esquema da figura 1) seria um programa de aplicação que gera, a partir de um conjunto de dados, uma figura que, depois de sofrer as transformações necessárias, é apresentada em um dispositivo de saída gráfico ativo (aquele que permite a interação). O operador, com base nesta saída, toma a decisão (de alterar, apagar, gerar segmentos, etc.) e novamente, através de outro dispositivo de entrada, informa-a ao sistema. O programa, então, efetua as correções necessárias, apresenta o resultado e o processo continua até não haver mais nenhuma correção.

Os micros e seus recursos gráficos

A evolução da microinformática veio permitir aos seus usuários um maior acesso aos recursos oferecidos pela computação gráfica. Os micros têm, quando comparados aos sistemas maiores, restrições inerentes ao hardware como baixa resolução gráfica e menor capacidade de memória, o que gera redução do número de cores possíveis de serem apresentadas. Exemplos de micros com maiores recursos gráficos e capacidade de gerar telas gráficas de alta resolução, com um número de cores razoáveis, são os micros de 16 bits. Já os micros das linhas Apple e TRS-80 Color, possuem menos opções de cores e baixo poder de cálculo, trazendo uma lentidão no processo de geração de imagens. Um outro fator que ajuda a limitar ainda mais os recursos gráficos dos micros, principalmente em termos de mercado brasileiro, é a disponibilidade de programas voltados a aplicações de comunicação visual.

Porém, mesmo com estas limitações, os micros estão sendo usados, cada vez mais, em aplicações que exijam transmissão de informações através de imagens. Um exemplo é o projeto desenvolvido por uma empresa fornecedora dos serviços de videotexto-Telesp na área de análises clínicas. A Video Soft, desenvolveu um pacote de software que permite ao usuário dos micros Apple acessar o sistema videotexto, gerando as páginas gráficas no próprio micro através do acoplamento de uma placa RS232. O micro, segundo a empresa, permite uma maior agilidade na apresentação das informações veiculadas — o que no videotexto (baseado no sistema francês) é muito lento.

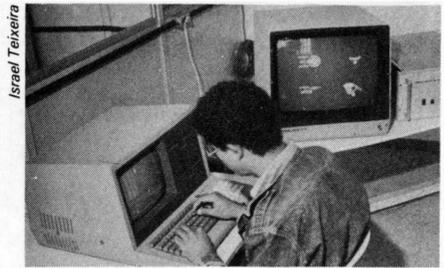
Nos Estados Unidos a microcomputação gráfica é muito utilizada, principalmente em publicidade, logicamente pela ampla disponibilidade de software. Lá, existem inúmeras opções de programas como por exemplo, um com algoritmos tridimensionais que produzem excelentes imagens de paisagens, tocadas à velocidade de três quadros por segundo.

A Experiência da Universidade na Área

O Laboratório de Subsistemas Integráveis ligado ao Departamento de Engenharia de Eletricidade da Escola Politécnica da USP iniciou há dois anos, um projeto de construção de um terminal gráfico "full graphics" tipo varredura, com acesso ponto a ponto, que agora começa a ser conhecido. O projeto foi coordenado pelo professor Antonio Zuffo e atualmente já existem algumas empresas interessadas em comercializá-lo, entre estas, a Edisa e a Digicom.

O terminal é compatível tanto com minicomputadores como com micros através da ligação de uma placa de comunicação RS-232 e, segundo o professor Zuffo, não teria um custo muito alto (em comparação com os já existentes), pois ele pode ser usado com uma TV comum, ficando em torno de Cr$ 5 milhões. Usando monitor comum — de tela entrelaçada — o pessoal do professor Zuffo está trabalhando atualmente com 520 X 380 pontos que podem ser expandidos até 640 X 400 linhas, pois atualmente, o terminal gráfico está usando um monitor de vídeo comum e apresenta algumas distorções nas bordas. A sua memória interna é associativa, permitindo a escolha de 4096 padrões de cores diferentes numa base de 1024 X 512 X 2 pontos. Usando uma placa de memória, o terminal da POLI pode gerar 4 cores; usando três placas permite 64 cores e assim por diante. O "full graphics" está permitindo, atualmente, a utilização de três planos independentes, sendo que cada plano possue quatro cores. Mas, com uma memória completa, o terminal poderia trabalhar com 256 cores simultaneamente.

Juntamente com o terminal gráfico foi desenvolvido também todo um pacote de software, compatível tanto com os terminais como com os plotters da Tektronics. A equipe gráfica do professor Zuffo está estudando no momento, dois padrões básicos de software: o GKS e o Kore — ambos derivados da Tektronics e que permitem a geração de imagens tridimensionais. Segundo o coordenador, as bases de softwares desenvolvidas visam principalmente as áreas de aplicativos com vistas à microeletrônica e a projetos mecânicos de engenharia.

No momento, Zuffo está iniciando um projeto maior, ligado ao "full graphics", que pretende desenvolver toda uma família de terminais com diversas capacidades e prevê ainda, a construção de estações gráficas para projetos de engenharia. Num outro plano, o grupo da Poli está executando um outro projeto de maior envergadura: um supermicro com 32 bits, que tem uma capacidade de três a quatro vezes superior ao VAX-780, e possue multiprocessamento e multiprogramação, contando ainda com software para ambientes Circs.

Israel Teixeira

Terminal gráfico da POLI/USP

A realidade brasileira da computação gráfica

Atualmente, conforme explanou o professor Zuffo, existem poucos usuários dos sistemas gráficos, assim como é inferior — quando comparado a outros países, o índice de projetos desenvolvidos nas universidades para este campo. Existem, segundo ele, um terminal full graphic (o que possue maior capacidade de resolução no mercado brasileiro), fabricado pela NCE do Rio de Janeiro e um outro fabricado pela Videotec em São Paulo. "Quanto aos softwares" — acrescentou — "em Campinas — São Paulo, existem pessoas desenvolvendo software graphics e também na Universidade Federal do Rio Grande do Sul, pelo grupo de pós-graduação de processamento de dados."

Quanto aos usuários, o professor citou empresas grandes como a Itautec e Scopus que estão trabalhando — embora paulatinamente — com projetos de implementação de recursos gráficos e softwares que melhorem a resolução gráfica de seus equipamentos.

Antonio Zuffo acredita que, pelo fato de os terminais gráficos serem uma forma eficiente de comunicação entre o usuário e a máquina, a longo prazo, eles ocuparão lugares semelhantes aos ocupados pelos terminais de vídeo.

A Experiência dos Usuários

Os dois maiores usuários de computação gráfica utilizam os sistemas comercializados pela Intergraph, representante no Brasil, da tecnologia Digital. Ambas, tanto Itautec como Scopus, estão entrando na área, sorrateiramente, e se preocupam principalmente com o software gráfico para seus equipamentos.

A Itautec trabalha hoje em dois eixos: um projeto de circuitos impressos — CAD (projeto auxiliado por computador) e software de gráficos administrativos, voltados à decisão.

A empresa pesquisa a área de computação gráfica desde 1980, demandando aproximadamente 100 pessoas por ano. Inicialmente, por volta de 1981, segundo Antonio Tadeu Mendes Beraldo, gerente da área de Sistemas e Processamento da Itautec, foi feito um projeto para gráficos administrativos padronizados, compatíveis com computadores IBM.

Este sistema recebia dados de um programa genérico chamando rotinas de desenho de qualquer ponto, através de um pós-processador (opcional nos programas de emissão de relatórios). Todo esse programa, ressalta Beraldo, não foi comercializado, ficando restrito ao uso interno do grupo Itaú.

Com o lançamento dos microcomputadores — a família I-7000 — foi desenvolvido um pacote contendo um traçador de gráficos administrativos padronizados, compatíveis com o sistema. Esse software ainda não foi lançado no mercado pois deverá, segundo Beraldo, receber ajustes, ganhando uma melhor resolução gráfica de tela, plotter e impressoras gráficas. Atendendo principalmente a área gerencial administrativa para apoio à decisão, o usuário desse pacote, através de um menu, escolhe entre vários tipos de gráficos disponíveis: barra, linhas, histogramas ou gráfico setorial, além de optar pelas legendas. Em termos de aplicações, são utilizados para síntese de relatórios numéricos gerenciais, tabelas de planos e avaliação de desempenho.

Projetos

Outra preocupação da empresa é o sistema CAD para projetos de placas de circuito impresso. Desde 1980, a Itautec trabalha num aplicativo que em sua fase inicial chamou-se versão 1 e que se caracterizou pelo desenvolvimento de placas para uso interno. Hoje esse projeto está em segunda fase, baseado na tecnologia da Intergraph. Nessa segunda versão, o projeto servirá eventualmente, segundo Beraldo, para prestação de serviços — transformando-se também num produto que fará parte da família I-7000 e do I-7000 PC além de outros produtos que a empresa pretende lançar.

A versão 2 deverá futuramente representar uma linha de pacotes, crescendo qualitativamente e se ligando a outros programas além da placa de circuito impresso. Ela inclui o desenvolvimento de módulos de apoio ao engenheiro projetista desse circuito, além de integrar os programas CAD (teste au-

tomático) para furação de placas e inserção de componentes. "Parte do CAD para circuito impresso", explica Beraldo, "que está sendo usado em sua versão 2, será oferecido mais tarde como um produto de mercado de bom nível, pois suas características de processamento distribuído, torna o sistema mais versátil".

Para ele, "a política da empresa hoje é investir no desenvolvimento de aplicativos, software gráfico básico e alguns equipamentos, aperfeiçoando o produto final antes de seu lançamento oficial no mercado".

Por isso foi desenvolvido, visando uso interno, um digitalizador para computadores de pequeno porte, em convênio com a FDTE (Fundação para o Desenvolvimento Tecnológico da Engenharia — USP). Outro projeto que se encontra em sua fase final é o chamado Núcleo Gráfico Básico padrão GKS, em convênio com a Fundcamp-Unicamp. Ele tem a finalidade de permitir que os comandos de processamento gráfico sejam vistos pelo programa de aplicação de forma padronizada e independente do dispositivo gráfico específico.

Servindo de suporte para todos os aplicativos de computação gráfica, outro projeto em desenvolvimento é o Desenhador de Propósitos Gerais (Grafiting). Por último, em termos de equipamentos, finaliza Beraldo, a família I-7000 será enriquecida pelo CAD Work Station — um micro de 32 bits — que a Itautec desenvolve em colaboração com o CTI (Centro de Tecnologia Informática).

Placas de circuito impresso: principal preocupação da Scopus

Começando a investir na área gráfica, a Scopus Tecnologia lançou recentemente, dois terminais remotos com alta resolução: o TVG-4001 e o TVA-3379 além de incentivar o desenvolvimento de softwares gráficos para seu microcomputador Nexus. Hoje a empresa está em fase de fechamento de contrato com a Intergraph para a produção de placas de circuito impresso.

O terminal remoto TVG-4001, projeta gráficos com até oito cores simultaneamente no vídeo, num total de escolha de 64 cores diferentes. Compatível com o Nexus, possui uma resolução gráfica de 320 pontos. O TVA-3279, atende a ambientes de equipamentos de grande porte, como o IBM 3279, trabalhando com gráficos programáveis.

Voltado ao mercado dos micros de 16 bits, o Nexus da Scopus recebeu um tratamento especial, segundo Nelson Lerner Barth, gerente de Teleprocessamento, na pesquisa de pacotes gráficos. As aplicações do equipamento procuram atender principalmente as áreas científicas, administrativas e de lazer — com jogos de alta precisão gráfica. Recentemente a Scopus firmou contrato com o IPT (Instituto de Pesquisas Tecnológicas) para o desenvolvimento de um pacote de desenho compatível com o Nexus e que pode opcionalmente receber o TVG como saída. Essas aplicações abrangem curvas de nível, histogramas tridimensionais, histogramas coloridos, histogramas com perspectivas, polígonos e superfícies. Diferentemente do plotter, o Nexus permite o aproveitamento da cor.

"Outra preocupação da Scopus", destaca José Moacyr Camargo, do departamento de apoio ao desenvolvimento, "é adquirir tecnologia de placas de circuito impresso através da Intergraph". Segundo ele, a introdução desse sistema reduzirá sensivelmente o tempo de execução do projeto de placas, agilizando a produção do lay-out ao fotolito e evitando a introdução de erros no projeto mecânico dos equipamentos fabricados pela empresa.

Entrevista Itautec e Scopus: Ana Luíza Mahlmeister

O Barão Vermelho na Computação Gráfica

A capa desta edição de Microhobby foi idealizada com o intuito de relembrarmos a sua edição número 0 que trazia a ilustração do barão Von Rietchentoff ou Barão Vermelho. Dessa forma, pensamos na visualização do famoso avião vermelho na tela de um computador. O equipamento escolhido, graças aos seus enormes e incríveis recursos gráficos, foi um VAX modelo mais antigo da Intergraph.

Inicialmente um de nossos colaboradores fez o projeto gráfico e depois, juntamente com um estagiário de engenharia eletrônica da empresa, este projeto passou a ser executado sistema-integraph. Segundo Roberto Bertini Renzetti (nosso colaborador) e o estagiário, Nilton Lobo Pinto, o tempo dispendido para a realização do desenho no VAX demorou aproximadamente dois dias. Para projetar o aviãozinho na tela os dois trabalharam com um arquivo em três dimensões. Primeiramente, projetou-se a asa num canto do vídeo, a fuselagem num outro e depois juntaram-se todas as peças. Detalhes como cor, ângulo e perspectiva foram definidos apenas depois de efetuada toda a montagem das peças. Conforme acrescentou Nilson, o desenho ocupou 2 MBytes de memória utilizando 4 cores com quatro mil blocos para formar a imagem do avião.

A Intergraph é uma das empresas líderes no fornecimento de sistemas gráficos para uso em mapeamento, engenharia, desenho urbano e arquitetura. A empresa fornece aos usuários de seus sistemas gráficos um pacote que é formado pelo hardware (processador central, uma ou mais estações gráficas, um console de controle on-line e arquivo de armazenamento dos subsistemas); o software, pela manutenção e treinamento. "A resolução gráfica de um VAX-750 — "afirmou o estagiário Nilton", está em torno de 4096 cores na memória principal (para os sistemas mais antigo) e 16 milhões — nos sistemas atuais — colocando 256 na tela do vídeo".

"O núcleo básico do sistema Intergraph" — acrescentou Nilton "é a estação gráfica de trabalho, composta de mesa digitalizadora, teclado, cursor, menu de comando e duas telas". É esse tipo de arranjo de telas duais que permite a visualização de uma visão espacial de um objeto tridimensional numa das telas e na outra, visões frontais, lateral e posterior do mesmo objeto facilitando métodos de trabalhos iguais aos usados na confecção da imagem do avião do barão vermelho — montando as partes em separado e remontando-as posteriormente.

Terminal gráfico (Work Station) com a unidade de processamento VAX 11/780

Impressão de caracter ao dobro da altura

Como uma boa dica para você, apresentamos um programa que modifica o formato dos caracteres contidos na ROM do TK.

Ao serem impressos, os caracteres estão o dobro de sua Altura Normal.

A impressão das palavras não é feita de uma só vez e sim em duas partes.

Como a listagem do programa é curta e sem grandes detalhes técnicos não há maiores problemas no momento da digitação.

```
   4 POKE 16389,124
   5 FOR I=0 TO 112
   6 POKE 31744+I,PEEK (I+2161)
   7 NEXT I
   8 POKE 31800,63
   9 POKE 31857,201
  90 DIM A$(32,16)
 100 LET B$="     MICROHOBBY - MI
CROHOBBY     "
 110 FOR Y=1 TO 32
 120 FOR X=0 TO 15
 130 LET A$(Y,X+1)=CHR$ PEEK (76
80+8*CODE B$(Y)+INT (X/2))
 140 NEXT X
 150 NEXT Y
 160 FOR I=0 TO 10 STEP 8
 170 FOR J=1 TO 32
 180 FOR K=1 TO 8
 190 POKE 32255+K+8*(J-1),CODE A
$(J,K+I)
 200 NEXT K
 210 NEXT J
 220 FOR H=0 TO 31
 230 POKE 16444+H,H
 240 NEXT H
 245 LET HP=USR 31744
 250 NEXT I
```

Impressora Mônica de Elebra

Este mês estamos inaugurando uma nova seção: Analisando.

Aqui investigaremos equipamentos, software e periféricos existentes no mercado.

O equipamento escolhido para inaugurar a seção é a impressora Mônica, fabricada pela Elebra Informática.

Teste: Caio M. Bulhões

Foto: Israel Teixeira

Montagem utilizada para teste: TK 2000, software para uso de impressora, programa de cadastro (Multicad), cabo para impressora, formulário contínuo sem pauta e formulário com etiquetas.

Impressoras são geralmente equipamentos caros, fora do alcance da maioria dos usuários de computadores pessoais. Entretanto, algumas delas se encontram numa faixa de preços da mesma ordem de grandeza que alguns computadores mais populares. São impressoras leves, geralmente com apenas 80 colunas, baixa velocidade, mas que atendem muito bem a uma boa parte das aplicações da maioria dos usuários. Entre elas está a Mônica, fabricada pela Elebra Informática.

Características técnicas

A Mônica é uma impressora de impacto, com agulhas distribuidas numa matriz de 9 x 7, com uma velocidade de 100 cps, com busca lógica de melhor caminho (ou seja, quando o carro acaba de imprimir o último caractere de uma linha, ele passa imediatamente para a próxima, sem percorrer desnecessariamente os espaços em branco até o final da linha).

Como equipamento opcional existe uma placa gráfica especial destinada a imprimir gráficos em alta resolução.

O padrão de impressão é o Anadex, o que a diferencia da maioria das impressoras nacionais, que adotaram o padrão Epson (ou Centronics). Devido a isto, aparecem alguns problemas, sobretudo no uso de gráficos, quando se tenta rodar alguns software desenvolvidos para o outro padrão, a menos que sejam inseridas algumas modificações.

A Mônica possui um seletor tipográfico com o qual se pode escolher entre um caractere normal e outro comprimido, fornecendo ao usuário uma opção a mais na impressão de seu material.

Condições de teste

Utilizamos em nossos testes o TK 2000 da redação, um TK 85 provido com interface para impressora e um compatível com o Apple. A Mônica fornecida não possuia placa gráfica.

Para que o TK 2000 pudesse ser usado com impressora, tivemos que primeiro carregar um software desenvolvido pela Multisoft para esta finalidade. Este software é fornecido em fita juntamente com a capa conectora que permite interfaciar o TK 2000 com impressoras de entrada paralela, como é o caso da Mônica. Mesmo com este Software, o TK 2000 não apresenta os comandos LPRINT e LLIST. Ao invés disso, toda vez que tivermos que usar a impressora, devemos ativá-la ou desativá-la por meio das instruções CALL 1302 (ativar) e CALL 1305 (desativar), que acessam sub-rotinas em linguagem de máquina disponíveis no programa que carregamos.

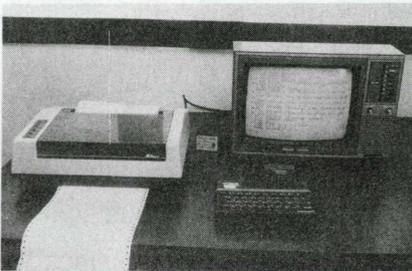



Foto: Israel Teixeira

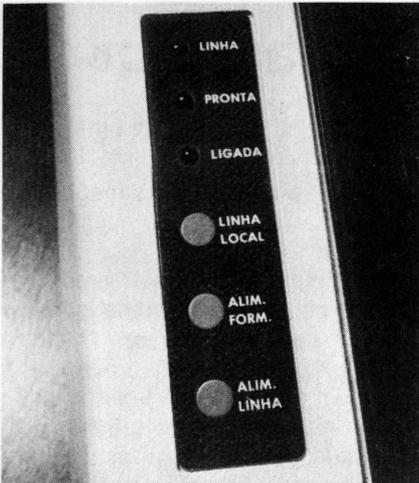
Montagem utilizada para teste: TK 85, interface para impressora, programa Editor de Cartas, formulário contínuo sem pauta.

Foto: Israel Teixeira

Interface para impressora: permite ao TK 85 ser ligado a qualquer impressora que utilize como padrão o Código ASCII.

O teste da impressora com este software em particular foi satisfatório, com ambos os caracteres, havendo apenas um problema: não conseguimos que a impressora imprimisse listagens com menos de 80 colunas de limites, apesar do programa permitir, através de POKEs, que pudéssemos alterar o colunamento. Consultamos a Microdigital a respeito e nos foi informado que isto era devido ao fato da Mônica utilizar-se do padrão Anadex, revelando-se aí uma pequena listagem.

Como toda impressora, a Mônica dispõe de um conjunto de letras minúsculas. Para obtê-las no TK 2000 devemos ativar os caracteres gráficos especiais, o que é feito pressionando-se as teclas CONTROL e B simultaneamente. Após este procedimento, pressiona-se SHIFT mais a letra ou conjunto de letras que se quer imprimir. O que aparecerá na tela não são letras minúsculas, mas sim caracteres gráficos especiais, cujos códigos são usados pela impressora co-

Foto: Israel Teixeira

Detalhe do painel: quando o LED superior estiver aceso, indica que podemos enviar dados do computador. O segundo LED indica quando o carro foi posicionado. O terceiro LED indica se a alimentação está ligada. O botão marcado LINHA LOCAL, ao ser precionado, bloqueia o envio de dados pelo computador e permite ao usuário acionar os outros dois botões: Alimentação de Formulário (ALIM. FORM.), que permite ao usuário avançar uma linha e Alimentação de Linha (ALIM. LINHA), que permite ao usuário avançar uma linha.

mo letras minúsculas. Os resultados obtidos foram satisfatórios.

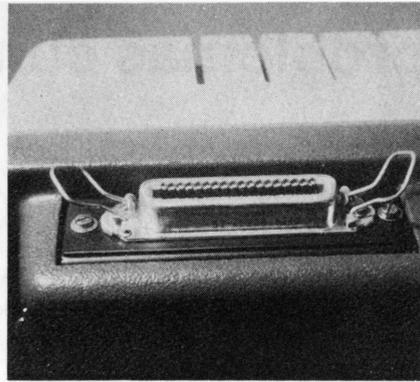
Além disso testamos a Mônica com um programa comercial, o Multicard, um programa para cadastramento onde dispomos da capacidade de emissão de relatórios e etiquetas. A impressora da Elebra apresentou um bom desempenho em ambas as tarefas.

Teste com o TK 85

Para esta fase, tivemos que lançar mão de uma interface especial, fabricada pela Microdigital, que permite o interfaciamento do TK 83 e 85 com uma impressora que possua entrada paralela e que aceite o código ASCII, como é o caso da Mônica. Como software de apoio utilizamos o programa Editor de Cartas, publicado na Microhobby número 6.

A interface utilizada traduz o código de caracteres do TK para o padrão ASCII. Isto é feito sacrificando-se os caracteres gráficos especiais, que são usados como caracteres controle da impressora. Desta maneira, apenas os caracteres alfanuméricos normais e os gráficos dos comandos PLOT e UNPLOT podem ser impressos normalmente quando se quer um COPY da tela. Os caracteres gráficos especiais causarão mudanças de linha, alimentação de formulário, ou outro comando de impressora. As letras invertidas (Grafics + letra) serão usadas como minúsculas. Os testes feitos tiveram resultados satisfatórios.

Foto: Israel Teixeira

Cabeça impressora: a Mônica possui uma cabeça impressora com agulhas, capacitada a imprimir caracteres normais e expandidos por meio de uma matriz de 9 x 7 pontos.

Entrada do cabo: a entrada de dados é feita por meio de um cabo de 16 vias paralelas.

O teste com o Apple

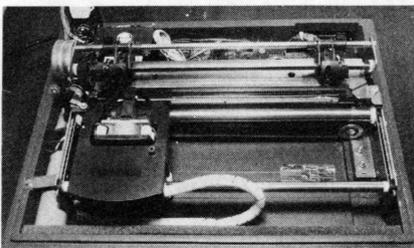
Utilizamos um compatível com o Apple rodando um programa em disco chamado VISIFILE, que permite a emissão de relatórios e etiquetas. Deixamos a Mônica durante várias horas, emitindo etiquetas e relatórios, sem que fosse registrado nenhum problema digno de nota.

As listagens de programas também foram bem realizadas, num desempenho semelhante ao do obtido junto ao TK 2000.

As letras minúsculas puderam ser também obtidas sem problemas, já que o compatível com o Apple que dispúnhamos apresentava esta opção (também para tela).

Por dentro e por fora

A abertura da Mônica revelou uma montagem simples, que facilita a manutenção. A colocação dos componentes indica um cuidado projeto do lay-out do circuito impresso e da distribuição interna, facilitando a montagem e a manutenção.

Foto: Israel Teixeira

O interior da Mônica (1)

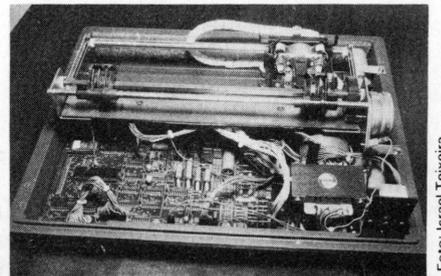
Detalhe da chave de troca de tipos de caracteres: permite ao usuário usar os caracteres compridos (chave na posição inferior) ou normais (chave na posição superior).

A fita está bem colocada, o que permite a troca rápida e segura. Os botões e comandos estão bem localizados, com excessão da chave de mudança de caracteres que está junto com o fusível e o fio de alimentação, sem qualquer indicação da sua finalidade ou para que lado devemos movê-la para obter um ou outro tipo de caractere, se bem que isto esteja bem explicado no seu manual de operações. Tivemos alguns problemas com o botão de alimentação de linha que se prendeu algumas vezes durante a operação.

O design é bonito, mas notamos que na tampa acrílica encontramos dois pontos onde ela nos pareceu frágil. O formulário passa através de um buraco feito nesta tampa, deixando frágeis os dois extremos. Acreditamos que a Elebra poderia melhorar isto sem prejuízo do design da Mônica.

Conclusão

Achamos que a Mônica é uma boa impressora dentro de sua faixa de tamanho e preço, executando bem diversas tarefas, além de possuir capacidade de trabalhar com gráficos. Seu tamanho permite uma instalação fácil e o seu design possui linhas agradáveis. Apesar dos problemas apontados, não deixa de ser uma boa opção ao usuário que deseja uma impressora com qualidade e preço acessível.

Foto: Israel Teixeira

O interior da Mônica (2): uma análise interna da Mônica revela um bom aproveitamento do espaço interno, sem sacrifício da manutenção. A disposição dos componentes foi feita de tal maneira a permitir o seu fácil acesso.

Os velhos problemas com o gravador

Tanios Hanzo

II PARTE

Nesta segunda parte darei mais algumas dicas de como usar, da melhor forma, o conjunto gravador-computador; mostrarei como identificar, prevenir e sanar qualquer mal-função por parte do gravador, sempre considerando o leitor como um leito em reparos técnicos e não equipado com as devidas ferramentas de um profissional.

Todo gravador deve ser ajustado segundo um padrão, para que uma fita gravada em qualquer um deles possa ser usada em qualquer outro. Este padrão é um dos requisitos para que um gravador cassete possa ser fabricado, exigido pela Philips holandesa, que inventou o sistema cassete em 1964.

A figura 5 representa a aparência de um cabeçote de gravador cassete comum. Note os diversos padrões que podem ser infringidos pelo desajuste de um simples parafuso. Em gravadores de boa qualidade, existem três parafusos prendendo o cabeçote à mesa; em gravadores de qualidade razoável, existem dois; e em gravadores ruins, só um parafuso. A importância destes parafusos não é apenas prender ou ajustar determinado padrão, mas também MANTER A REGULAGEM e permitir o acesso de ferramentas para reajustes, quando necessário.

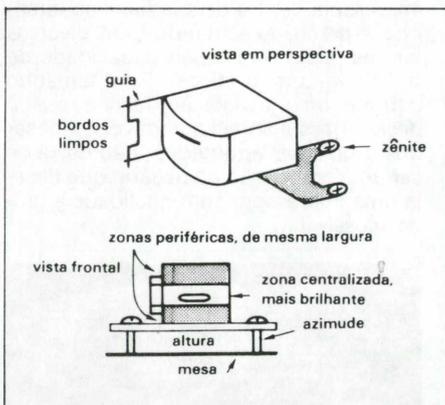



Fig. 5

Como um médico, aquele que quiser manter seu gravador com saúde de ferro (ou cromo) pessoalmente, deverá examinar o "paciente" à procura de sintomas para identificar a doença. Então deverá remediar o problema, corrigindo o mal e finalmente, prevenir o paciente contra abusos e descuidos posteriores.

Identificação Malfunções:

O método usado é o mais óbvio: deixe o paciente falar. A grande maioria dos problemas que podem ocorrer a um gravador, podem ser percebidos simplesmente ouvindo-o. Eis um exemplo de procedimento:

Põe-se para tocar uma fita com sons agudos presentes. Pode ser uma fita contendo música (na mais intuitiva percepção de sons pelo ouvinte) ou uma fita com programas gravados (que são tipicamente agudos, mas de percepção menos intuitiva), o importante é que possam ser identificados sons de alta freqüência (os sons agudos) com boa intensidade. Se o gravador possuir um controle de tonalidade, este deve ser ajustado de forma a realçar os sons mais estridentes — os sons agudos. Não é errado considerar que a qualidade de um gravador é diretamete proporcional à sua capacidade de reproduzir sons agudos e o mesmo pode ser aplicado às fitas magnéticas, no que diz respeito à gravação de programas.

Pois bem; se a fita ao ser reproduzida, não indicar a presença dos sons agudos e apresentar um chiado (hissss . . .), há um problema.

O paciente deve ser examinado. Abre-se a tampa do gravador e pressiona-se a tecla PLAY (sem querer parodiar o tradicional — Abre a boca e diga ahhhhh . . .). Se o acesso visual ao cabeçote for difícil, pode-se montar um "espelhinho", similar aqueles que os dentistas usam, esquematizando na figura 6, o que facilitará observar a aparência do cabeçote.

Um cabeçote sadio deverá ter sua face limpa e brilhante, sem qualquer marca.

Cataloguei oito casos comuns de "doença" em gravadores, e os apresento um a um, mostrando como verificá-los, recuperá-los e prevení-los.

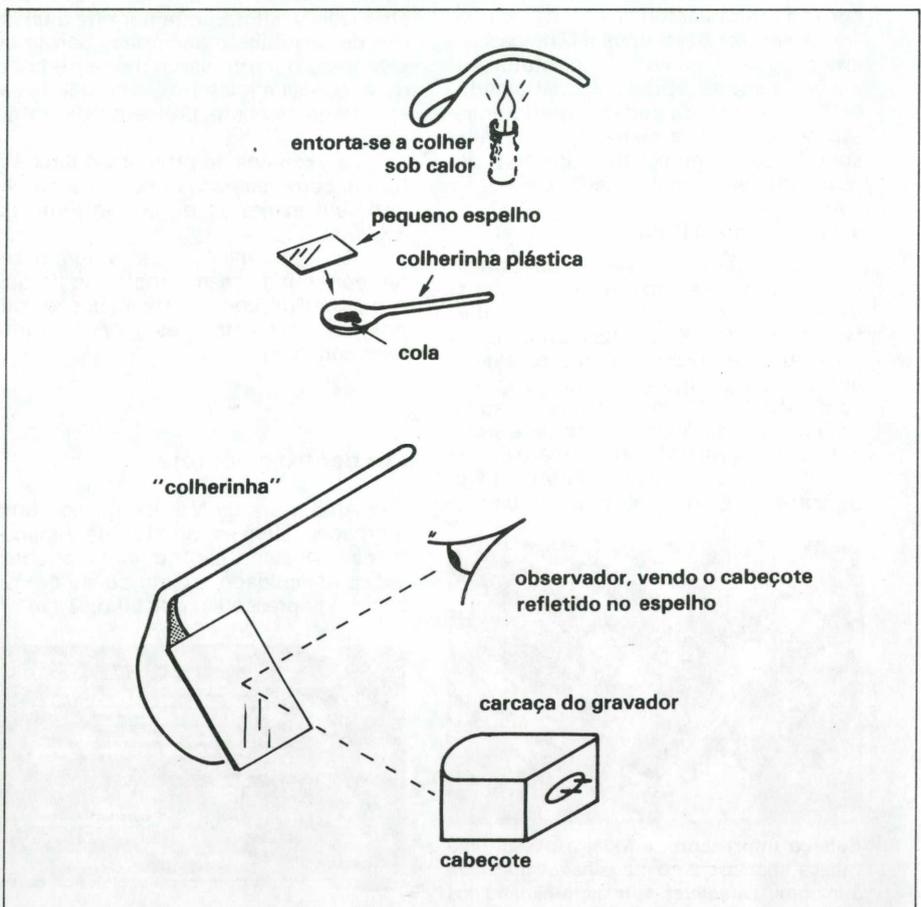



Fig. 6

CASO A — Altura:

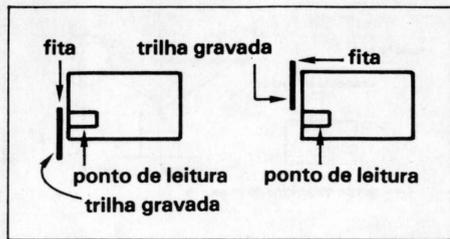



Fig. 7

A figura 7 ilustra este caso, que pode ser compreendido ao se comparar o ponto do cabeçote que lê a informação da fita a um carro, e a fita a uma pista única de uma estrada; o carro só deverá seguir alinhado à pista, do contrário, sairá da estrada. O alinhamento do cabeçote é feito ainda na fábrica, e assim deverá permanecer. Há casos, porém, em que um gravador desregulado escape ao controle de qualidade do fabricante ou o usuário o desregule inocentemente. A figura 5 mostra quais são os parafusos encarregados de ajustar e manter firme a altura correta, tensionados por molas e/ou espaçadores. Um gravador com cabeçote fora de padrão pode ser usado durante anos, sem que se perceba seu problema, pois funciona perfeitamente reproduzindo fitas gravadas por ele mesmo com perfeição. Com fitas gravadas em outros gravadores, padronizados, porém, a reprodução é deficiente, denunciando a anomalia. Observando o cabeçote suspeito, procure indícios que comprovem sua suspeita. SÓ MODIFIQUE A ALTURA DO CABEÇOTE SE TIVER PLENA CERTEZA DE QUE ESTÁ REALMENTE DESALINHADO, do contrário piorará a situação, remediando um problema que não existe e criando assim um novo.

Um cabeçote deselinhado tem a aparência do mostrado na figura 8. Com o cabeçote limpo, a zona brilhante (pela ação da fita deslizando sobre ele) não está alinhada como mostra a figura 9, o esquema de um cabeçote corretamente ajustado em altura.

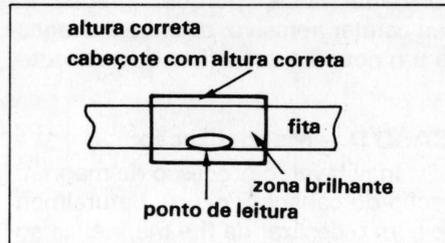



Fig. 9

Em casos de comprovado desajuste, procure saber se o cabeçote deverá ser levantado ou abaixado e verifique para que lado deverão ser girados os parafusos de ajuste de altura para corrigir o alinhamento, antes de colocar uma fita-padrão (uma fita com agudos gravados em um gravador com cabeçote comprovadamente padronizado) e prosseguir, girando primeiramente o parafuso de ajuste exclusivo de altura (no esquema da figura 5, o da esquerda) e depois o outro parafuso, em busca do maior volume possível do som, o que comprova o perfeito ajuste.

Importante: neste ajuste ou em qualquer outro, jamais use uma chave de fenda ou outra ferramenta imantada. O campo magnético de uma ferramenta imantada pode magnetizar o cabeçote. Uma ferramenta magnética é identificada se atrair limalha de ferro (ou pequenas farpas de bombril) e pode ser desmagnetizada, bastando aquecê-la numa chama e dando-lhe uma martelada ou pancada forte.

CASO B — Zênite:

A figura 10 ilustra o caso em que há o desajuste do zênite de um cabeçote. Note que neste caso, de nada vale estar o cabeçote alinhado em altura se não há o perfeito contato entre ele e a fita.

Nesta região que separa a fita do cabeçote desajustado, haverá menor atrito, o que implicará em contrastes acúmulos de sujeira, enquanto que na outra parte, onde o atrito entre a fita e o cabeçote é maior, haverá um maior desgaste do mesmo, além da deformação permanente da fita, que por ser elástica, acabará ficando com uma lateral enrugada e deteriorando para sempre a informação nela contida.

O procedimento para ajustar o zênite, se comprovado o seu desajuste, deverá ser similar ao descrito para o ajuste da altura, também sendo necessário girar o parafuso de zênite até o necessário para se obter uma saída alta, em termos de volume sonoro.

Muitos gravadores não possuem este parafuso, e o ajuste de um cabeçote assim só será possível com muito trabalho, desentortando o suporte do cabeçote onde existem os outros parafusos.

CASO C — Limpeza:

É o problema mais comum. É inevitável o acúmulo de partículas da própria fita ou de poeira no cabeçote de um gravador.

Ao usuário, resta limpar periodicamente, à cada 20 horas de uso, aproximadamente, para fitas boas. Se o leitor insistir em usar fitas de qualidade duvidosa, a limpeza deverá ser feita com maior freqüência.

Existem no mercado kits de limpeza de cabeçote, contendo o necessário para o trabalho: Um líquido para limpeza, uma pequena esponja de feltro ou espuma e um espelhinho.

O usuário poderá, como muitos já o fazem, limpar caseiramente o cabe-

Fig. 8

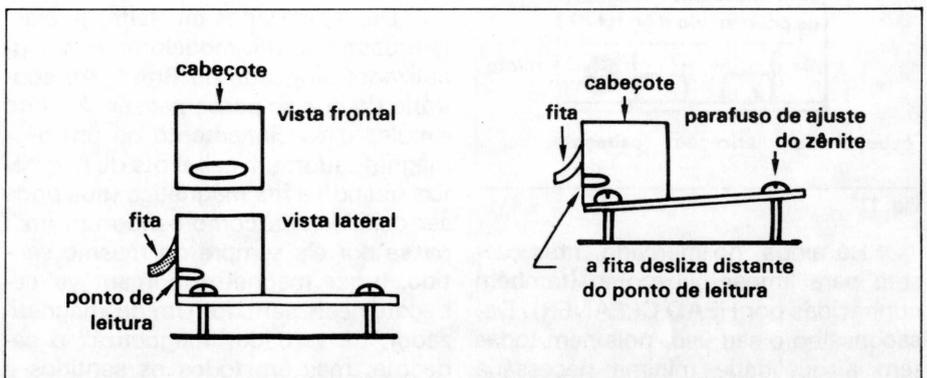



Fig. 10

çote, desde que NÃO USE POLIDORES para limpar metais, como o kaol, silvo, brasso, etc., que tem ação corrosiva penetrante, o que não desgasta o cabeçote externamente, como também penetra e corroi o entreferro e até a bobina. O uso de álcool comum para a limpeza do cabeçote também não é aconselhável, dada a sua característica de oxidar metais. Teoricamente não se deve usar produtos que possam reajir quimicamente com os materiais de que podem ser feitos os cabeçotes (ferrite, permalloy, etc.), porém se este produto for volátil (evaporar com facilidade) o suficiente para não permitir nenhuma reação significativa, seu uso pode ser admitido. Pode-se usar álcool isopropílico, por exemplo.

O líquido de limpeza deverá ser aplicado por uma esponja qualquer, desde que não seja atacada pelo líquido e não desprensa fragmentos. O uso de cotonetes deve ser feito de forma cuidadosa, de modo a não permitir o desprendimento de farpas, que podem ser muito finas e penetrarem parcialmente pela "janela" do cabeçote, o que seria um desastre, porque arranharia toda a extensão de uma fita cassete, arrancando-lhe partículas e sujando rapidamente o cabeçote.

Já que se está com a mão na massa, pode-se limpar também o cabeçote de polarização (aquele lateral) e o rolete de borracha quando se limpar o cabeçote. Limpeza nunca é demais . . .

A figura 11 mostra como um cabeçote sujo prejudica a perfeita ação do conjunto fita-cabeçote, e a disposição da mesma, em relação ao rolete e ao cabeçote de polarização.

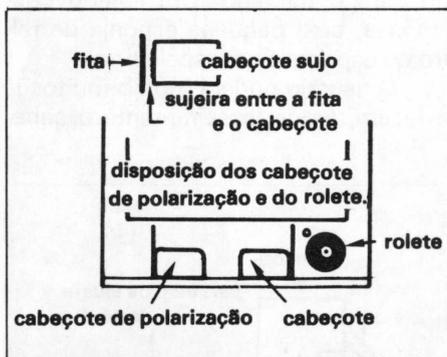



Fig. 11

Há ainda, no mercado, fitas cassete para limpar cabeçotes (também conhecidas por HEAD CLEANER). Desaconselho o seu uso, pois nem todas têm a qualidade mínima necessária para não remover mais do que partículas de sujeira. A fita de limpeza geralmente é feita de um papel especial, muitas vezes importado. Se o usuário encontrar alguma fita de limpeza de comprovada qualidade, vale a pena adquirí-la mas não deverá ser usada constantemente, porque, graças ao seu caráter abrasivo, arranha e desgasta um pouco a superfície do cabeçote.

CASO D — Magnetização:

Inevitável, o processo de magnetização do cabeçote ocorre naturalmente com o deslizar da fita magnética sobre o cabeçote, que é magnetizável. O uso de fitas incompatíveis com gravadores acelera o processo de magnetização, que não só proporciona uma péssima reprodução como também deteriora a informação gravada na fita magnética.

A identificação de um cabeçote magnetizado não pode ser feita pela simples observação. Um cabeçote magnetizado não é visível, mas seus efeitos são perceptíveis ao se ouvir uma fita: os sons agudos são muito baixos; há chiado, o som é abafado e distorcido.

Imperceptível, mas presente, há também problemas na gravação, já que o cabeçote fica magnetizado mesmo sem haver uma informação para ser gravada. Um cabeçote já bem magnetizado se comporta como um cabeçote de polarização, apagando o conteúdo da fita que por ele passar, permanentemente.

Pouca gente já ouviu falar em magnetização dew cabeçote. Menos gente ainda ouviu falar em DESMAGNETIZADORES de cabeçote, a única maneira de remediar o problema.

A figura 12 mostra esquematicamente um cabeçote magnetizado e um desmagnetizador em uso.

Depois de anos em falta, já existe atualmente um modelo de desmagnetizador fabricado no Brasil. Ao contrário do que se possa pensar, é muito simples o funcionamento de um desmagnetizador: um cabeçote de magnetiza quando a fita magnética (que pode ser considerada como sendo um imã) passa por ele sempre no mesmo sentido, o que magnetiza o imantável cabeçote neste sentido. Um desmagnetizador, na verdade "magnetiza" o cabeçote, mas em todos os sentidos e por igual, o que resulta em uma magnetização nula.

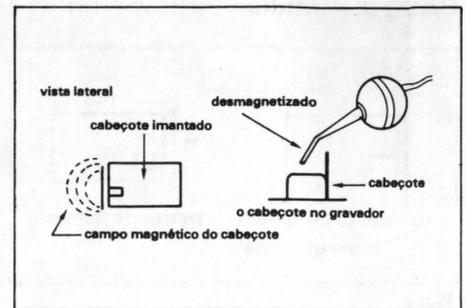



Fig. 12

Embora mais raros, existem desmagnetizadores eletrônicos aqui no Brasil (todos importados, infelizmente), que não dão muita mão-de-obra e têm funcionamento garantido, independente do usuário.

Mais raros ainda, são os limpadores/desmagnetizadores, (cordless cassete head cleaner and demagnetizer) que limpa e desmagnetizam em um minuto qualquer cabeçote, sem pilhas e sem caráter abrasivo, disponível infelizmente, só no exterior.

Computação McGraw-Hill

LIVROS DE QUALIDADE

INICIAÇÃO AO BASIC
Fox/Fox — Cr$ 9.000,00

Escrito em estilo fácil, este livro, destina-se especialmente aos principiantes que não tenham acesso aos micros, mas que desejam familiarizar-se com os conceitos de programação.
Pode ser aplicado a qualquer computador que use a linguagem BASIC.
Um texto alegre que aceita a possibilidade do leitor não ter experiência anterior.

CP/M — GUIA DO USUÁRIO
Hogam — Cr$ 13.900,00

Escrito para usuários de bom nível de conhecimento. Este livro considera a história e funções do CP/M bem como os comandos próprios para o usuário. Inclui o CP/M-86, sistema operacional baseado no 8086 e 8088.

INTR. AOS MICROCOMPUTADORES
Osborne — Cr$ 10.400,00

Livro para principiantes em microcomputação. Os conceitos básicos sobre todos os Micros: como funcionam e o que eles podem fazer. Introduz o leitor nas linguagens de programação, códigos binários e aritmética, lógica, temporização, memória e como usá-los.

APPLE II — GUIA DO USUÁRIO
Poole — Cr$ 22.900,00

Este é o melhor e mais completo manual do APPLE II. Contém descrição de todas instruções, comandos e funções.
Uma seção especial em programação avançada e aplicações. Claro e objetivo. Obrigatório para os usuários do APPLE II.

PROGRAMAS USUAIS EM BASIC

* **Programas usuais em Basic TRS-80**
* **Programas usuais em Basic APPLE II**
* **Programas práticos em Basic**

Cr$ 10.800,00 cada

* Vários programas práticos para pequenos negócios, pequenas empresas de Engenharia, Administração, Matemática e Economia Doméstica.

MANUAL DE BASIC PARA O APPLE II
Peckham — Cr$ 11.700,00

Manual prático que possibilita ao leitor aprender a programar o APPLE II através de exercícios dirigidos.
Escrito em linguagem fácil, acessível, dirigido a hobistas e estudantes.

VISICALC
Castlowitz — Cr$ 12.900,00

Um guia prático para utilização do software VISICALC. Através da leitura deste manual, o usuário poderá obter o máximo em qualidade e eficiência em sua atividade.

CONSTRUA SEU PRÓPRIO MICROCOMPUTADOR
Usando Z-80
Ciarcia — Cr$ 21.900,00

Este guia prático, mostra como você pode construir seu próprio microcomputador, baseado no famoso microprocessador, o ZILOG Z-80.
Cada subsistema do computador é completamente explicado e baseado em informações testadas de forma que o leitor possa facilmente modificar o sistema.
Muito fácil compreensão.

PROGRAMAÇÃO TKS 82-83-85 CP-200
Hurley — Cr$ 4.900,00

Aprenda a programar seu TK e CP-200 muito facilmente.
Programas em BASIC, jogos, gráficos, etc., para principiantes.
Fácil assimilação e compreensão.

MICROPROCESSADORES — Conceitos Básicos
Osborne — Cr$ 11.000,00

A mais compreensiva e atualizada introdução ao sistema de microprocessadores.
Tudo sobre microprocessadores.
A maneira mais fácil e simples de entender os microprocessadores.
Conceitos básicos.

NOVA REVOLUÇÃO INDUSTRIAL
Na Era dos Computadores
Osborne — Cr$ 5.400,00

Esta convincente e provocativa obra, fornece ao leitor, esclarecimentos sobre o poder e o futuro dos micros, antecipando mudanças radicais até o final deste século.
O autor explica a revolução da microeletrônica e seu impacto na sociedade.

LIVROS UNIVERSITÁRIOS E CURSOS

COMPUTADORES E PROGRAMAÇÃO
Gottfried — Cr$ 11.200,00

A finalidade deste texto é fornecer um curso de programação de computadores, empregando a estrutura padronizada da linguagem BASIC.
Proporciona ao leitor condições de organizar e escrever eficientemente programas de computador. 350 problemas resolvidos. Destina-se a todos o curso do ensino Superior e Técnico que utiliza o BASIC como linguagem.

PROGRAMAÇÃO COM BASIC
Scheid — Cr$ 11.300,00

Quinhentos e trinta e cinco problemas resolvidos e vários programas usando a linguagem FORTRAN, BASIC, PASCAL e PL/1.
Introdução à Computação e Programação, destinado ao mercado Universitário em cursos de Engenharia e Administração.

CIÊNCIA DOS COMPUTADORES
Tremblay — Cr$ 11.900,00

Introdução à Ciências dos Computadores, usando uma abordagem ALGORITMICA.
Livro texto dirigido à Engenharia, Matemática e outras áreas afins.
Best Seller nos EUA.

INTRODUÇÃO A CIÊNCIA DOS COMPUTADORES
Scheid — Cr$ 9.000,00

Teoria que possibilita plena compreensão da introdução à ciência dos computadores.
Indicado para cursos de Engenharia e Administração (2ºs anos).
Vários problemas propostos e resolvidos.

CIRCUITOS DIGITAIS E MICROPROCESSADORES
Taub — Cr$ 13.400,00

Livro texto dirigido à Escolas de Engenharia, às áreas de projetos lógicos e microprocessadores.
Concentram-se nos modelos 8080/8086 com detalhes para aplicações em outras unidades. Explicações claras sobre FLIP-FLOPS e MEMÓRIAS.
300 problemas cuidadosamente elaborados.
UM BEST SELLER!

PROCESSAMENTO DE DADOS — Volume I
PROCESSAMENTO DE DADOS — Volume II
Verzello — Cr$ 9.300,00 cada

Mantendo-se independente de qualquer tipo de linguagem ou tipo específico de máquina, os autores discutem todos os assuntos da área de maneira integral.
Como usar a tecnologia de computação para resolver problemas de processamento de dados.
Livro introdutório para Economia, Administração e Engenharia. É acessível a estudantes com limitados conhecimentos matemáticos.

EDITORA McGRAW-HILL DO BRASIL LTDA.

NOME ______________________________

ENDEREÇO ___________________________

LIVROS ______________________________

____

____

ANXEXO CHEQUE ☐ REEMBOLSO POSTAL ☐

Rua Tabapuã, 1105 - CEP 0453 - Telefone: 881-8604 - Itaim-Bibi - São Paulo - Brasil.

*** Em todas livrarias do Brasil ou diretamente. Solicite catálogos.**

RESPOSTA DO QUEBRA CABEÇA

Renato da Silva Oliveira

"A PERIGRINAÇÃO DO DALAI LAMA"

```
   1 REM A PEREGRINACAO DO DALAI
 LAMA
   2 REM DOUGLAS RENAUX
   3 FAST
   5 LET MIN1=2000
   6 LET MIN2=MIN1
   7 LET C=0
   8 LET D$=""
  10 FOR I=2 TO 7
  20 FOR J=2 TO 7
  30 IF J=I THEN GOTO 180
  40 FOR K=2 TO 7
  50 IF K=I OR K=J THEN GOTO 170
  60 FOR L=2 TO 7
  70 IF L=I OR L=J OR L=K THEN G
OTO 160
  80 FOR M=2 TO 7
  90 IF M=I OR M=J OR M=K OR M=L
 THEN GOTO 150
 100 FOR N=2 TO 7
 110 IF N=I OR N=J OR N=K OR N=L
 OR N=M THEN GOTO 140
 120 LET DIST=A(1,I)+A(I,J)+A(J,
K)+A(K,L)+A(L,M)+A(M,N)+A(N,1)
 130 IF DIST>=MIN1 THEN GOTO 139
 131 LET MIN1=DIST
 132 LET C$="1"+CHR$ (I+28)+CHR$
 (J+28)+CHR$ (K+28)+CHR$ (L+28)+
CHR$ (M+28)+CHR$ (N+28)+"1"
 133 LET E$=C$
 134 IF DIST>=MIN2 THEN GOTO 139
 135 LET MIN1=MIN2
 136 LET MIN2=DIST
 137 LET C$=D$
 138 LET D$=E$
 139 LET C=C+1
 140 NEXT N
 150 NEXT M
 160 NEXT L
 170 NEXT K
 180 NEXT J
 190 NEXT I
 191 GOTO 400
 200 FOR I=1 TO 7
 210 FOR J=1 TO 7
 215 INPUT A(I,J)
 220 PRINT TAB (J-1)*4;A(I,J);
 250 NEXT J
 260 PRINT
 265 NEXT I
 270 GOTO 3
 400 PRINT "OS DOIS CAMINHOS MAI
S CURTOS      SAO:",,C$;"  DISTAN
CIA=";MIN1;" KM",D$;" . DISTANCIA
=";MIN2;" KM"
 410 PRINT "NUMERO DE ALTERNATIV
AS TENTADAS ";C

 T=3 MIN 50 S
 MEM.=2394 BYTES
```

Com grande surpresa e satisfação, recebemos em tempo hábil para publicação três soluções para o quebra-cabeça "A Peregrinação do Dalai-Lama". Mesmo considerando que esse problema clássico é de amplo conhecimento e utilidade, não esperávamos receber tão já soluções corretas, dado o modo como o problema foi proposto.

Os três programas recebidos são semelhantes entre si e, coincidentemente, semelhantes também ao que o sr. Nabor utilizou. O critério para a escolha do que estamos publicando baseou-se como de costume, em dois aspectos:

1) A quantidade de memória utilizada.
2) O tempo utilizado no processamento do programa.

Desse modo, o programa de Douglas Renaux saiu-se melhor, pois roda em cerca de 3 min e 50 s e ocupa aproximadamente 2kBytes de RAM.

O programa enviado por Zilda G. Plassa ocupa cerca de 3kBytes e demora pouco mais de 7 min para rodar. O programa de Marcel G. de Albuquerque, na forma como nos chegou às mãos, suas informações são previamente analisadas antes de serem processadas e, mesmo assim, demora pouco mais de 4 m para rodar e ocupa pouco menos de 2kBytes na RAM. Sem desmérito algum para o Marcel e para a Zilda, escolhemos o programa do Douglas.

A Solução

Apresento aqui a solução encontrada para o quebra-cabeça da revista nº 9, "A peregrinação do Dalai Lama".

O programa consiste na geração das 720 alternativas de trajeto e na medida da distância percorrida em cada trajeto, memorizando os dois trajetos mais curtos. Além disso é feita uma contagem do número de alternativas.

As localidades foram numeradas de 1 a 7 da seguinte maneira:

1 — Selipuk; 2 — Barka; 3 — Menze; 4 — Rajem; 5 — Gartok; 6 — Thok Jalung; 7 — Thok Daurakpa.

As distâncias estão numa matriz A(I,J) que deve ser fornecida pelo usuário antes de rodar o programa. Para tal basta: DIM A(7,7) GOTO 200.

O programa vai preenchendo a matriz, começando pelo canto superior esquerdo da tabela dada e continuando pela 1ª linha . . ., deve-se colocar 0 onde aparece — na tabela.

Terminando a tabela, o programa começa a rodar automaticamente.

As duas alternativas são:

1 7 4 2 3 5 6 1
1 6 5 3 2 4 7 1

ambas com 1006 Km

Notar que uma é o percurso inverso da outra.

O caminho escolhido é o primeiro, pois o trecho com mais montanhas e lagos é Selipuk — Thok Daurapko — Rajem (174) que deve ser feito no início.

OBS: O programa e as variáveis ocupam pouco menos de 1500 bytes.

Solução:
SELIPUK
THOK DAURAPKA
RAJEM
BARKA
MENZE
GARTOK
THOK JALUNG
SELIPUK

Introduzindo o "Ampersand"

Parte I

Coordenação:
Prof. Wilson José Tucci
Colaborações:
Daniel Roberto Falconer
José Eduardo Moreira

A capacidade de interfaceamento com rotinas escritas em Linguagem de Máquina é característica comum na maioria dos dialetos do BASIC, e o Applesoft não é exceção. De maneira semelhante às outras versões de linguagem, o BASIC do Apple permite a chamada direta de rotinas em Linguagem de Máquina, tanto através do comando *CALL* como no da função *USR.* No entanto, o Applesoft possui um comando raramente aproveitado, a não ser através de utilitários comerciais: o comando &.

Este comando (denominado "Ampersand", em inglês), possui uma característica muito importante. Normalmente, ele não faz absolutamente nada. Porém, quando preparado adequadamente, o Ampersand pode dar-nos uma conexão direta com praticamente qualquer rotina em Linguagem de Máquina dentro do Apple, tanto as nossas quanto as que se encontram embutidas dentro da linguagem e do DOS. E isso não se limita a apenas um comando. Com um pouco de bom senso e criatividade (além de conhecimentos de Linguagem de Máquina), qualquer um pode adicionar inúmeras rotinas de máquina, executando tarefas tão variadas quanto ordenar rapidamente uma matriz alfabética, produzir uma listagem das variáveis definidas na memória ou transferir uma tela de gráficos para a impressora. E tudo isso pode ser acessado no modo direto, ou dentro de um programa BASIC, como se essas rotinas em Linguagem de Máquina fossem parte integrante do BASIC.

O objetivo desta primeira parte não é ensinar a programar em Linguagem de Máquina; o leitor deve ter algumas noções para poder acompanhar algumas partes destes artigos. No entanto, esta primeira parte pode ser entendida sem esses conhecimentos. Veremos aqui como fazer o interfaceamento básico com rotinas existentes (ou feitas pelo leitor); na próxima parte serão dadas as noções de como definir mais de um comando baseado no &.

O Vetor &

Sempre que o comando & é executado no BASIC, este deixa de lado temporariamente o que estiver fazendo, para executar a rotina de Linguagem de Máquina localizada na posição $3F5, ou 1013 decimal (todos os endereços no sistema hexadecimal serão dados com o símbolo '$' precedendo o número), de forma semelhante ao modo como são executadas sub-rotinas em BASIC.

A partir desta posição estão reservados três bytes, que geralmente são usados colocando-se um *JMP* (comando de desvio em Assembly), ao início da rotina que desejamos ver ativada. Como dissemos acima, normalmente o Ampersand não faz nada, isto porque esse "vetor" (o desvio incondicional), aponta para uma instrução *RTS* (equivalente ao *RETURN* do BASIC), retornando ao ponto onde foi encontrado o comando & e continuando como se nada tivesse ocorrido.

O Comando BASIC *CATALOG*

Quando queremos fazer com que o & ative uma rotina qualquer, simplesmente devemos colocar, na posição $3F5, um *JMP* ao endereço inicial dessa rotina. Vamos, como exemplo, ligar a Ampersand à rotina de DOS que mostra o diretório do disco. O início do comando *CATALOG* está na posição $A56E (ou 42350) *. Colocar um *JMP $A56E* na posição $F35 é tudo o que devemos fazer. Olhando em uma tabela das instruções de máquina do 6502, podemos codificar esse desvio por

$4C $6E $A5 (no sistema hexadecimal)

ou

76 110 165 (no sistema decimal)

Podemos *"POKEar"* esses números (a partir da posição 1013), com o comando:

```
POKE 1013, 76: POKE 1014, 110:
POKE 1015, 165
```

ou através do monitor:

```
CAL — 151 (para entrar no monitor)
3F5: 4C 63 A5
(para colocar os valores)
3F5L
(aparece o JMP $A56E no topo da tela)
3D0G (volta ao BASIC)
```

Agora, um simples & (seguido, é claro, do RETURN), produzirá o catálogo do disco. Perceba que & é um comando BASIC e, portanto, pode ser colocado dentro de um programa, sem a necessidade de colocá-lo dentro de um *PRINT* (como é comumente feito com os comandos de DOS). Ele também pode ser usado no meio de um comando, separado por dois pontos, como sempre fizemos com os outros comandos Applesoft.

Outros Exemplos

Há centenas de rotinas dentro do Apple que podem ser chamadas como &; no entanto, por enquanto, estamos limitados àquelas que não exigem parâmetros de entrada ou interfaceamento mais complexo. Vejamos algumas.

Para os que usam muito o monitor, o Ampersand pode substituir o tão comum *CALL — 151.* O endereço — 151 é equivalente ao decimal positivo 65385, ou $FF69 no sistema hexadecimal. Os códigos que devem ser colocados nas posições 1014 e 1015 (assumindo que 1013 já contém o JMP), são, respectivamente, 105 e 255 (ou $FF e $69, a partir de $3F6). Use um dos modelos acima. Terminado, o Ampersand ativará o monitor, como fazíamos antes com o *CALL.*

O mesmo pode ser feito com os comandos *LIST* ou *RUN* do BASIC. A linha

```
POKE 1014, 165: POKE
POKE 1014, 165: POKE 1015, 214
```

ou

```
3F6: A5 D6 (no monitor)
```

fará com que o Ampersand aja como o *LIST.* Da mesma forma,

```
POKE 1014, 102: POKE 1015, 213
```

ou

```
3F6: 66 D5
```

executará um *RUN* quando for dado o comando &.

Uma aplicação que pode ser muito útil para aquelas rotinas de "PRESSIONE UMA TECLA PARA CONTINUAR." O vetor do Ampersand pode ser mudado para apontar à rotina do monitor que espera por uma tecla, sem o cursor. Basta *"POKEar"* os valores 27 e 253 ($1B e $FD), como fizemos acima. Isso pode ser feito no início de um programa e este pode usar diretamente o comando & quando quiser esperar por uma tecla qualquer.

Um último exemplo: faça

```
POKE 1014, 166: POKE 1015, 250
```

e execute o &. Para que serve? Se alguém tiver alguma utilidade para este exemplo, realmente gostaríamos de saber . . .

Conclusão

Apenas raspamos a superfície deste comando tão ignorado. Esperamos, nos próximos números, poder introduzir algumas formas um pouco mais avançadas de utilizar o Ampersand. Por enquanto, o melhor é experimentar, lembrando sempre que, acontecendo qualquer problema, sempre podemos desligar o computador. Apenas tenham cuidado para deixar um disco importante no drive ao experimentar com endereços pertencentes ao DOS!

---

* *Nota: qualquer endereço de DOS acima de 40000 decimal refere-se a sistemas com, pelo menos, 48K de memória; subtraia 16384 para Apples com 32K, ou mais 16384 para aqueles com 16K.*

Por dentro do Apple
Errata

A codificação completa do programa Disco Voador, publicado nos números 11 e 12, segue abaixo:

LISTAGEM A

```
0  REM  ©1984 POR DENTRO DO APPLE

   DISCO VOADOR

100  GOSUB 980: HOME : INVERSE
110 PA$ = "  @#&!  "
120 Q =  - 16384:BU = Q + 48:G$ =
     CHR$ (7)
130  GR
140  VTAB 23: HTAB 1: PRINT ":DIS
    COS::";: HTAB 11: PRINT "::T
    IROS::";: HTAB 21: PRINT ":A
    CERTOS:";: HTAB 31: PRINT ":
    :::%::::"
150  VTAB 24: HTAB 1: PRINT "::::
    :::::";: HTAB 11: PRINT ":::
    ::::::";: HTAB 21: PRINT "::
    :::::::";: HTAB 31: PRINT ":
    ::00%:::";
160  GOSUB 580:SL = 35:XF = 1:YY =
    0:HR = 0:CD =  INT ( RND (1)
     * 14) + 1: GOSUB 940
170  VTAB 21: HTAB 1
180  PRINT "<ESPACO> ATIRA, OUTRA
     PARA A BALA"
190  GOSUB 940
200  REM ===CANHAO===
210 CC =  INT ( RND (1) * 14) + 1
    : IF CC = CD THEN 210
220  COLOR= CC: HLIN 22,26 AT 39:
     HLIN 23,25 AT 38: VLIN 36,3
    7 AT 24
230 K =  PEEK (Q): IF K <  > 160 THEN
    410
240  IF SL <  > 35 THEN 270
250 HD = HD + 1: GOSUB 840
260  GOSUB 900
270  COLOR= 15
280  IF  SCRN( 24,SL - 1) = 0 THEN
    380
290  REM ===ACERTOU!===
300  PLOT 24,SL - 1: COLOR= 0: PLOT
    24,SL
310  COLOR= 9
320  PLOT 22,SL - 3: PLOT 23,SL -
    2: PLOT 23,SL: PLOT 24,SL -
    1: PLOT 25,SL: PLOT 27,SL -
    2: VLIN SL - 3,SL - 2 AT 25
330  FOR I = 1 TO 9: GOSUB 850: NEXT
    :AC = 1
340  COLOR= 0: FOR I = 22 TO 27: VLIN
    SL - 3,SL AT I: NEXT
350 HI = HI + 1: GOSUB 900
360 YY = 1:XF = 0
370  POKE Q + 16,0: GOSUB 430
380  PLOT 24,SL - 1: COLOR= 0: PLOT
    24,SL
390 SL = SL - 1: IF SL > YF - 3 THEN
    410
400  COLOR= 0: VLIN 0,35 AT 24:SL
     = 35: POKE Q + 16,0
410  GOSUB 430
420  GOTO 230
430  REM ===DISCO===
440  COLOR= CD
450  HLIN HR + 2,HR + 3 AT YF: HLIN
    HR + 1,HR + 4 AT YF + 1
460  COLOR= 0
470  PLOT HR + 1,YF: PLOT HR,YF +
    1
480  IF AC THEN  PLOT HR + 4,YF: HLIN
    HR,HR + 5 AT YF - 1: HLIN HR
    ,HR + 5 AT YF - 2
490  IF YY = 0 THEN 530
500 N = 41 - YF: GOSUB 950
510 XF = XF +  INT ( RND (1) * 3)
     - 1: IF HR + XF < 19 THEN X
    F = 1
520  IF HR + XF > 25 THEN XF =  -
    1
530 HR = HR + XF:YF = YF + YY: IF
    HR > 35 THEN 560
540  IF YF > 38 THEN  GOSUB 600
550  RETURN
560  HLIN 36,39 AT YF + 1: HLIN 3
    7,38 AT YF
570  GOSUB 940:CD =  INT ( RND (1
    ) * 14) + 1: IF CD = CC THEN
    570
580 AD = AD + 1: GOSUB 900
590 HR = 0: RETURN
600  REM ===EXPLOSAO===
610  VTAB 21: HTAB HR: PRINT PA$:
    AC = 0
620  COLOR= 15
630  PLOT HR,37: PLOT HR + 1,38: VLIN
    35,37 AT HR + 2: PLOT HR + 2
    ,39: VLIN 38,39 AT HR + 3: VLIN
    36,37 AT HR + 4: PLOT HR + 5
    ,37
640  POP : PRINT G$;G$;G$;G$
650  COLOR= CD
660  FOR I = 39 TO 6 STEP  - 6
670  HLIN 19,29 AT I - 1
680  HLIN 19,29 AT I: GOSUB 870
690  NEXT
700  HLIN 29,33 AT 2: HLIN 20,23 AT
    3: HLIN 28,34 AT 3: HLIN 18,
    24 AT 4: HLIN 27,35 AT 4: HLIN
    17,35 AT 5: HLIN 16,35 AT 6:
     HLIN 15,35 AT 7
710  HLIN 16,36 AT 8: HLIN 13,37 AT
    9: HLIN 12,38 AT 10: HLIN 11
    ,38 AT 11: HLIN 11,38 AT 12:
     HLIN 10,38 AT 13: HLIN 10,3
    8 AT 14: HLIN 10,37 AT 15
720  HLIN 10,37 AT 16: HLIN 10,37
     AT 17: HLIN 11,14 AT 18: HLIN
    16,36 AT 18: HLIN 12,13 AT 1
    9: HLIN 16,34 AT 19: HLIN 16
    ,34 AT 20: HLIN 17,34 AT 21
730  HLIN 31,33 AT 22
740  FOR I = 1 TO 22
750  IF I -  INT (I / 2) * 2 = 0 THEN
     COLOR= CD
760  IF I -  INT (I / 2) * 2 = 1 THEN
     COLOR= CC
770  FOR J = 1 TO 2 * I: NEXT
780 HR =  INT ( RND (1) * 40):YF =
     INT ( RND (1) * 15) + 20: GOSUB
    870: PLOT HR,YF: NEXT
790  COLOR= 0
800  FOR I = 0 TO 39: HLIN 0,39 AT
    I: FOR J = 1 TO 9: NEXT : NEXT
    : FOR J = 1 TO 222: NEXT
810  POP : GOTO 160
820  FOR I = 1 TO 500: NEXT
830  REM ===BARULHO===
840  COLOR= 9: HLIN 23,25 AT 35: HLIN
    22,23 AT 34: HLIN 25,26 AT 3
    4
850  FOR J = 1 TO 10:S =  PEEK (B
    U) -  PEEK (BU) -  PEEK (BU)
     -  PEEK (BU) -  PEEK (BU): NEXT

860  COLOR= 0: HLIN 23,25 AT 35: HLIN
    22,26 AT 34: RETURN
870  FOR J = 1 TO 2
880 S =  PEEK (BU) -  PEEK (BU) -
     PEEK (BU) -  PEEK (BU) -  PEEK
    (BU): RETURN
890  REM ===PONTOS===
900  VTAB 24: HTAB 4: PRINT AD;: HTAB
    14: PRINT HD;
910  HTAB 24: PRINT HI;: IF  NOT
    HI THEN  RETURN
920  HTAB 34: PRINT  INT ((100 *
    HI) / AD);"%:";
930  RETURN
940 YF = Y1:Y1 =  INT ( RND (1) *
    20) + 10: RETURN
950  POKE 6,100 - (3 * N): POKE 7
    ,5
960  CALL 771
970  RETURN
980  FOR I = 771 TO 789: READ A: POKE
    I,A: NEXT : RETURN
990  DATA 173,48,192,136,208,4,19
    8,7,240,8,202,208,246,166,6,
    76,3,3,96
```

TWO-LINERS

Mais um two-liner de gráficos. O programa deste mês nos foi enviado por Ricardo Orsi, de São Paulo. Esta, como quase todas as colaborações que recebemos, produz formas interessantes na tela. Apenas um comentário: o Ricardo aconselha usar, na hora de executar o programa, uma altura de 30 ou mais e 3 ou 4 ciclos, para gerar uma boa tela. É isso aí! Enviem suas colaborações para:
Revista Microhobby
Seção Two Liners
Caixa Postal 54096
01296 — São Paulo — SP

```
0  REM *** PROGRAMA POR
          RICARDO ORSI
1  TEXT : HOME : INPUT "CICLOS: "
     ;C: INPUT "ALTURA: ";A: IF C
      > 4 OR A > 70 THEN 1
2  HGR2 : FOR I = 0 TO 6.28 * C STEP
     .05:G = G + .5: HPLOT 279,0 TO
     G, SIN (I) * A + 80: HCOLOR=
     0: HPLOT 270,0 TO G - .5, SIN
     (I) * A + 79: HCOLOR= 3: NEXT
     : GET A$: GOTO 1
```

Fig. 1

Fig. 2

Por dentro do Apple
Errata

Na revista 8, à página 27, onde lê-se:
800 FOR TH = 1 TO T2 STEP PASSO
leia-se:
800 FOR TH = T1 TO T2 — PASSO STEP PASSO

Perspectiva exata para o TK 2000

Vívian Bernardo

A computação gráfica é um dos possíveis usos que podemos dar ao microcomputador, desde que ele tenha recursos para isso.

O microcomputador TK-2000, da Microdigital é um equipamento capacitado para trabalhar em computação gráfica, porque possui, de forma residente, instruções e comandos que permitem trabalhar-se em alta resolução, para a obtenção de gráficos, desenhos e figuras geométricas na tela.

Os modos de operação

Para podermos trabalhar em alta resolução, devemos trabalhar no modo de operação adequado. O TK-2000 possui três diferentes modos de operação:

— **modo texto:** iniciado pela instrução TEXT, de baixa resolução gráfica; a tela é definida com 24 linhas horizontais e 40 linhas verticais. As instruções usadas no modo texto para formatação de tela são: PRINT, TAB, VTAB, HTAB.

— **modo normal de baixa resolução gráfica:** iniciado pela instrução GR; a tela é dividida em 48 colunas e 40 linhas (valores de 0 a 39 na horizontal e 0 a 47 na vertical). Uma vantagem desse modo de resolução gráfica é a possibilidade de obtermos gráfico e texto na mesma tela, já que as quatro últimas linhas da tela permanecem no modo texto. A instrução para desenhos no modo normal de baixa resolução é PLOT. Seu formato é: PLOT X,Y, onde X é a coordenada horizontal e Y a vertical. Esta instrução define pontos isolados na tela sendo mais usada no traçado de retas nos eixos bidimensionais da tela. Outras instruções usadas neste modo são:

— HLIN e VLIN, com formato HLIN O,Y TO X, sendo X a coluna fixada e Y o limite do traçado da reta;

— COLOR, que permite o uso de cores numeradas de 0 a 15 para reprodução de linhas coloridas na tela do micro (tabela I). Seu formato é COLOR = a (a é um número entre 0 e 15);

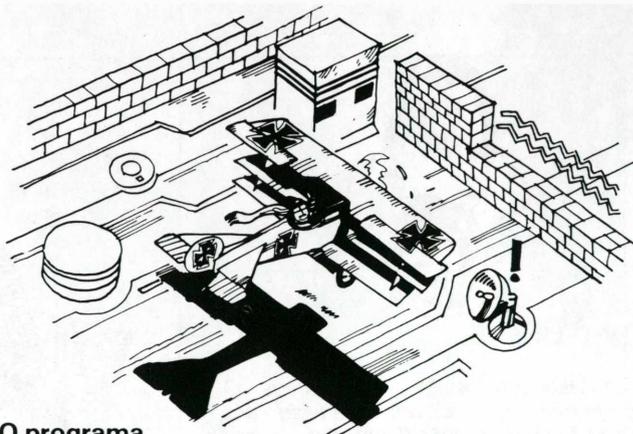
— SCRN: associa um ponto da tela ao número da cor correspondente, por exemplo:

```
10 HOME: GR: COLOR = 6
20 VLIN 0,20 AT 20: PRINT SCRN (20,20): END
```

Você terá uma linha no sentido vertical na coluna 20 em cor azul.

— **modo de alta resolução gráfica:** iniciado pela instrução HGR. Através dela, você pode controlar cada ponto da tela que passa a ter dimensão de 280 pontos horizontais e 160 pontos verticais. Também neste modo de resolução gráfica, as linhas finais da tela permanecem em *modo texto.*

A instrução para construção de figuras em alta resolução é HPLOT. Sua função é análoga à instrução PLOT, somente com a diferença de um maior número de pontos definidos na tela. Seu formato é: HPLOT X1, Y1 TO X2, Y2, onde X1, Y1 e X2, Y2 são pontos quaisquer pelos quais será traçada uma linha reta. Há também o formato HPLOT X1, Y1 que "plota" em alta resolução, um ponto isolado na tela.

Caso você queira eliminar as quatro linhas finais no modo texto, associe a instrução HGR2 ao programa e sua tela passará a ter 280 pontos horizontais e 192 verticais. Desta forma você estará usando a segunda página de vídeo.

O programa

O programa Perspectiva Exata foi elaborado para demonstrar os recursos de computação gráfica que o TK-2000 oferece. Ele constrói quatro figuras em perspectiva, usando modo de alta resolução gráfica com a instrução HPLOT para a "plotagem" na tela.

Há uma rotina para a configuração de cada uma delas (GOSUB 2000, GOSUB 3000, GOSUB 4000, GOSUB 5000). Você pode observar que, no início de cada rotina, foi colocada a instrução HGR, para que fosse possível usar o comando HPLOT. O formato usado é:

HPLOT X1, Y1 TO X2, Y1 TO X2, Y2 TO X1, Y2 TO X1, Y1.

Desta forma, pode-se obter um retângulo com uma única instrução. Nos endereços 2030, 3020, 4020, 5020 temos uma malha com FOR . . . NEXT, onde o limite é o número de retângulos que constituem a figura. Os dados das instruções DATA, em cada uma das rotinas, são as coordenadas dos pontos de cada retângulo.

Na instrução HCOLOR, foi colocado o número 7 (cor branca) para todas as figuras. Por que? Porque a cor branca "plota" em todos os pontos da tela de alta resolução, o que não ocorre com as outras cores disponíveis no TK 2000.

Para que você possa entender a causa da restrição no uso das cores, é necessário um bom conhecimento do mapa de memória de alta resolução gráfica. O byte de geração de cor tem o primeiro bit, chamado bit de cor, que controla a cor, ou seja, determina em qual grupo de cores serão apresentados os 7 bits restantes. Quando o bit de cor é 0, são apresentadas as cores PRETO 1, AZUL, CYAN e BRANCO 1 pertencentes ao grupo 1.

Para o bit de cor igual a 1, são mostradas as cores PRETO 2, VERMELHO, VERDE e BRANCO 2, do grupo 2. Desta forma, ao "plotarmos" BRANCO 1 na coluna 0, o bit 1 passa a ter valor 1 e o resto do byte fica nulo (incluindo o bit de cor); quando "plotamos" PRETO 2 na coluna 0, os bits da coluna 0 passam a ser 0 (já eram de qualquer forma) e o bit de cor passa a ter valor 1 provocando uma mudança do byte inteiro para as cores do grupo 2; então o azul, que é a próxima cor do grupo 1 passa a ser na verdade, a sua correspondente do grupo 2: vermelho.

Conclusão

As cores do grupo 1 podem ser usadas nas colunas ímpares e as cores do grupo 2, somente nas colunas pares. A cor branca é a única disponível em todas as colunas da tela de alta resolução, e por isso foi escolhida para este programa. Estudando-se corretamente as restrições para cores no TK-2000 (que também é válida para equipamentos como Apple, com interface para cor), colocando-se a cor certa no ponto certo da tela, você obterá trabalhos gráficos bastante satisfatórios.

TABELA I

Ø — preta	6 — cyan	11 — branca
1 — azul	7 — branca	12 — branca
2 — verde	8 — verde	13 — vermelho
3 — branca	9 — azul	14 — cyan
4 — azul	1Ø — verde	15 — branca
5 — vermelho		

PROGRAMA

```
]LIST

5  REM  MICROMEGA / SETEMBRO - 1984:RE
M AUTHOR : 'VIVIAN'
10  REM  PROGRAMA DE PERSPECTIVA/ALTA
RESOLUCAO/TK-2000/EDICAO N. 13
20  GOSUB 1000
100  GOSUB 2000
120  GOSUB 3000
140  GOSUB 4000
160  GOSUB 5000
180  TEXT : HOME : VTAB 11: HTAB 7: PRINT
"FIM": VTAB 13: HTAB 3: PRINT "MICRO
MEGA - COMPUTACAO GRAFICA": VTAB
15: HTAB 10: PRINT "SETEMBRO - 19
84"
200  END
1000  REM  ROTINA DE APRESENTACAO
1010  HOME : PRINT "ESTE PROGRAMA DEMO
NSTRA OS COMANDOS DE ALTA RESOLUC
AO GRAFICA DO TK-2000.": PRINT : PRINT
"APRESENTA QUATRO FIGURAS EM PERSP
ECTIVA:": PRINT : PRINT
1020  PRINT "- UM CUBO (UM QUADRADO EM
 PERSPECTIVA).": PRINT : PRINT "-
 UM CRUXIFIXO (DOIS PARALELOGRAMO
S EM PERSPECTIVA)
1030  PRINT : PRINT "- UMA ESCADA (SER
IE DE PARALELOGRAMOS EM PRSPECTIV
A).": PRINT : PRINT "- UMA PALAVR
A (COM PONTO DE PERSPECTIVA PARA
A DIREITA).": PRINT : PRINT
1040  PRINT "DIGITE 'C' PARA OBTER AS
TELAS DE CADA FIGURA"
1050  GET E$: IF E$ ( ) "C" THEN 1050

1060  RETURN
2000  REM  ROTINA DA CONSTRUCAO DO QUA
DRADO EM PERSPECTIVA
2010  HOME : HGR : HCOLOR  = 7
2030  FOR N = 1 TO 6: READ X1,Y1,X2,Y2
,X3,Y3,X4,Y4,X5,Y5: HPLOT X1,Y1 TO X2,
Y2 TO X3,Y3 TO X4,Y4 TO X5,Y5: NEXT
2040  GET C$: IF C$ ( ) "C" THEN 2040

2050  RETURN
2060  DATA  90,50,130,70,130,130,90,90
,90,50,90,50,170,39,170,67,90,90,
90,50,170,39,219,48,219,85,170,67
,170,39,130,70,219,48,219,85,130,
130,130,70,90,50,170,39,219,48,13
0,70,90,50,90,90,170,67,219,85,13
0,130,90,90
3000  REM  ROTINA DE CONSTRUCAO DA CRU
Z EM PERSPECTIVA
3010  HOME : HGR : HCOLOR  = 7
3020  FOR N = 1 TO 7
3025  READ X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,
Y5: HPLOT X1,Y1 TO X2,Y2 TO X3,Y3
 TO X4,Y4 TO X5,Y5: NEXT N
3030  GET C$: IF C$ ( ) "C" THEN 3030


3040  RETURN
3055  DATA  130,48,143,47,200,40,190,3
6,130,48,145,11,159,3,159,121,145
,112,145,11,130,37,130,48,190,36,
190,17,130,37,158,106,172,112,159
,121,145,112,158,106,190,17,200,2
9,200,40,190,36,190,17
3058  DATA  159,3,172,11,172,112,159,1
21,159,3,159,42,190,36,200,40,172
,45,159,42
4000  REM  ROTINA DE CONSTRUCAO DA ESC
ADA EM PERSPECTIVA
4010  HOME : HGR : HCOLOR  = 7
4020  FOR N = 1 TO 21
4025  READ X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,
Y5: HPLOT X1,Y1 TO X2,Y2 TO X3,Y3
 TO X4,Y4 TO X5,Y5: NEXT N
4030  GET C$: IF C$ ( ) "C" THEN 4030


4040  RETURN
4050  DATA  65,36,120,38,130,37,80,35,
65,36,65,36,120,38,120,96,65,105,
65,36,65,105,120,96,193,110,150,1
30,65,105,150,130,150,110,193,95,
193,110,150,130,150,110,128,105,1
74,92,193,95,150,110,128,105,128,
85,174,78,174,92,128,105
4055  DATA  128,85,110,82,156,76,174,7
8,128,85,110,82,110,67,156,63,156
,76,110,82,110,67,93,65,140,62,15
6,63,110,67,93,65,93,50,140,50,14
0,62,93,65,80,50,80,35,130,37,130
,50,80,50
4058  DATA  65,36,80,35,80,109,65,105,
65,36,80,50,93,50,93,113,80,109,8
0,50,93,65,110,67,110,118,93,113,
93,65
4059  DATA  120,38,130,37,130,98,120,9
6,120,38,130,50,140,50,140,100,13
0,98,130,50,140,62,156,63,156,103
,140,100,140,62,156,76,174,78,174
,107,156,103,156,76,174,92,193,95
,193,110,174,107,174,92
4060  DATA  110,82,128,85,128,124,110,
118,110,82,128,105,150,110,150,13
0,128,124,128,105
5000  REM  ROTINA DE CONSTRUCAO DA PAL
AVRA
5010  HOME : HGR : HCOLOR  = 7
5020  FOR N = 1 TO 40: READ X1,Y1,X2,Y
2,X3,Y3,X4,Y4,X5,Y5: HPLOT X1,Y1 TO X2,
Y2 TO X3,Y3 TO X4,Y4 TO X5,Y5: NEXT N
5030  GET C$: IF C$ ( ) "C" THEN 5030
5040  RETURN
5050  REM  DADOS LETRA M
5055  DATA  40,55,58,50,58,73,40,80,40
,55,40,55,58,50,65,40,55,65,40,55
,80,50,65,40,55,65,65,55,80,50,80
,50,65,55,65,80,80,73,80,50
5060  REM  DADOS LETRA I  E  C
5065  DATA  83,55,97,50,97,73,83,80,83
,55,100,55,112,50,112,72,100,80,1
00,55,100,55,120,55,130,50,112,50
,100,55,100,80,112,72,130,72,120,
80,100,80
5070  REM  DADOS LETRA R
5075  DATA  130,55,155,55,164,50,140,5
0,130,55,130,55,155,55,155,65,130
,65,130,55,155,55,155,65,164,58,1
64,50,155,55,130,65,140,58,164,58
,155,65,130,65,130,65,140,58,164,
71,155,80,130,65,130,55,140,50,14
0,72,130,80,130,55
5080  REM  DADOS LETRA O
5085  DATA  167,50,190,50,185,55,160,5
5,167,50,185,55,190,50,190,72,185
,80,185,55,160,55,160,80,185,80,1
85,55,160,55,160,55,167,50,167,72
,160,80,160,55,167,50,190,50,190,
72,167,72,167,50
5090  REM  DADOS LETRA H
5092  DATA  40,105,56,97,56,119,40,130
,40,105,65,105,79,97,79,119,65,13
0,65,105,40,118,56,108,79,108,65,
118,40,118
5095  DATA  65,105,95,105,106,97,83,97
,65,105,65,105,95,105,95,130,65,1
30,65,105,95,105,106,97,106,120,9
5,130,95,105,83,97,83,120,65,130,
65,105,83,97,83,120,106,120,95,13
0,65,130,83,120
5100  REM  DADOS DA LETRA B
5105  DATA  111,97,128,97,125,105,100,
105,111,97,100,117,111,108,128,10
8,125,117,100,117,100,130,111,120
,128,120,125,130,100,130,125,105,
128,97,128,120,125,130,125,105,11
1,97,111,120,100,130,100,105,111,
97
5110  REM  DADOS DA LETRA B(SEGUNDA)
5115  DATA  138,97,161,97,155,105,130,
105,138,97,130,117,138,108,161,10
8,155,117,130,117,130,130,138,120
,161,120,155,130,130,130,130,105,
138,97,138,120,130,130,130,105,15
5,105,161,97,161,120,155,130,155,
105
5120  REM  DADOS LETRA Y
5125  DATA  165,97,177,105,173,115,160
,105,165,97,177,105,189,97,185,10
5,173,115,177,105,173,115,177,105
,177,120,173,130,173,115
```

Relógio solar horizontal cálculo analítico no TK

Renato da Silva Oliveira

Os relógios solares são, provavelmente, os mais antigos instrumentos criados pelo homem para medir o tempo. Conhecidos há milhares de anos, foram largamente utilizados por quase todos os povos da Terra.

Há diversos tipos e montagens de relógios solares, entretanto todos baseiam-se num mesmo fato: o movimento aparente do Sol no céu.

Nosso estudo será dirigido aos *relógios solares horizontais,* que são os mais simples e comuns. Eles constituem-se basicamente de um anteparo plano no qual são marcadas as horas do dia e de um ponteiro que é afixado no anteparo (Baseie-se na figura 1 para acompanhar o texto). Normalmente, a posição das linhas horárias é obtida por processos geométricos. Aqui, vamos calculá-las por um método analítico, com auxílio do TK.

Inicialmente, deve-se construir o ponteiro, cujo nome mais correto é *veleta.* Para tanto, basta desenharmos um triângulo retângulo em que um dos ângulos agudos é igual à latitude do local onde será instalado o relógio. O triângulo será a veleta e seu cateto (representado por AB na figura 17 determinará as diménsões mínimas do anteparo. Uma vez construída a veleta deve-se passar à construção do anteparo, que será realizada com auxílio do micro.

Usualmente, no anteparo são marcadas apenas as horas desde as 5 da manhã até as 19 da noite. Em nosso programa, você poderá escolher entre o usual ou o cálculo dos minutos (de 10 em 10, 15 em 15 ou 30 em 30). Ao rodar o programa, você deverá introduzir a latitude do local onde o relógio será usado. Para auxiliá-lo na tabela I damos a latitude de várias localidades do Brasil.

A seguir, você deve introduzir de quanto em quanto tempo deverão ser calculadas as marcações do anteparo. A partir disso, o micro apresentará no vídeo uma tabela de três colunas. Na primeira estão as horas, na segunda os minutos e na terceira o ângulo α (alfa), formado entre cada linha horária e a linha horária central (correspondente às 12 horas). O ângulo cresce positivamente no sentido anti-horário e negativamente no sentido horário. Como exemplo, na figura 2 representa-se os ângulos α_1 e α_2 correspondentes às linhas horárias das 8 horas da manhão e das 19 horas da noite para a latitude 23°30' 00'', referente à cidade de São Paulo.

Após ter construído o relógio, você deve posicioná-lo corretamente. Faça isso, orientando a linha horária das 12 horas na direção Norte-Sul geográfica, como mostra a figura 3.

Finalizando, há duas observações a serem feitas:

1) O relógio solar calculado pelo programa *HOROLOGIUM* não leva em consideração as variações no movimento aparente do Sol no céu. Sendo assim, as horas por ele indicadas são geralmente aproximadas, atingindo valores bem perto das reais nos equinócios (instantes da passagem do Verão para o Outono e do Inverno para a Primavera).

2) Não tente usá-los sobre o Equador da Terra. Lá, o ângulo entre as linhas horárias é sempre nulo! Existem modelos de relógios solares mais convenientes para este local.

Relógio Solar Para o Hemisfério Sul da Terra. O ângulo φ é a Latitude do Local onde ele será instalado. Nas proximidades do Equador, as linhas horárias das 5 e das 19 horas são desnecessárias.

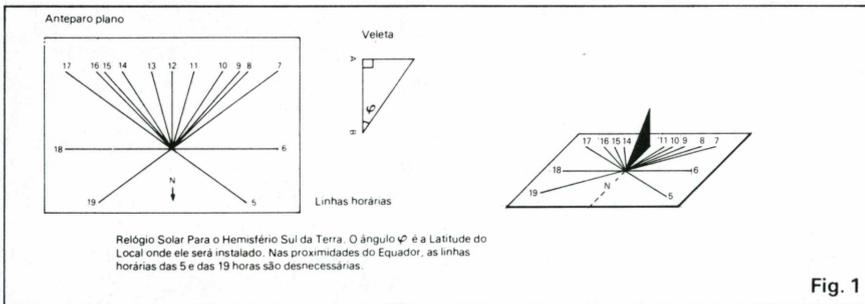



Fig. 1

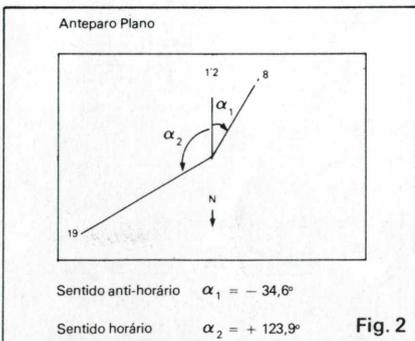



Fig. 2

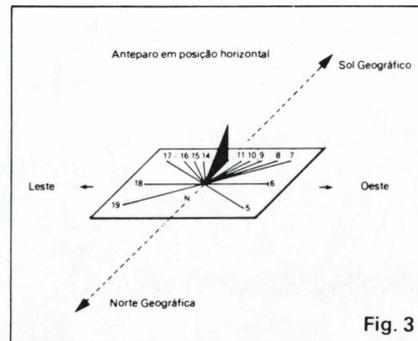



Fig. 3

TABELA I

Latitude de algumas cidades

Cidade	Latitude Sul
Belém	1° 10'
Manaus	2° 50'
Olinda/Recife	8° 5'
Salvador	13° 0'
Fortaleza	3° 40'
Cuiabá	15° 30'
Brasília	15° 45'
Campo Grande	20° 30'
Rio de Janeiro	22° 55'
São Paulo	23° 40'
Porto Alegre	30° 0'

TABELA II

Código de caracteres gráficos usados

Linha	Códigos/Caracteres/Usados
110	180
165	2 e 156
170	10, 2 e 3
400	10

```
   1 REM HOROLOGIUM
   5 LET I$="MINUTOS"
  10 PRINT "INTRODUZA A LATITUDE
:"
  15 PRINT
  20 PRINT "GRAUS=>";
  30 INPUT G
  40 PRINT G
  50 PRINT I$;"=>";
  60 INPUT M
  70 PRINT M
  80 PRINT "SEGUNDOS=>";
  90 INPUT S
 100 PRINT S
 110 LET G=SIN (PI*(G+(M+S/CODE
"W")/CODE "W")/CODE "■")
 115 IF G=PI-PI THEN GOTO VAL "4
00"
 120 PAUSE CODE "Z"
 130 CLS
 140 PRINT "ALFA DEVE SER CALCUL
ADO DE QUANTO EM QUANTO TEMPO?"
 145 PRINT
 150 PRINT "(1)10 EM 10 ";I$,"(2
)15 EM 15 ";I$,"(3)30 EM 30 ";I$
,"(4)UMA EM UMA HORA"
 160 INPUT M
 165 PRINT AT CODE "■"+M,PI/PI;C
HR$ (CODE "■"+M)
 170 LET S=CODE "■"*(M=PI/PI)+15
*(M=CODE "■")+CODE "2"*(M=CODE "
■")+CODE "W"*(M=4)
 175 LET I$="-"
 180 PAUSE 60
 185 CLS
 190 LET M=VAL "15300"
 200 LET N=VAL "21600"
 205 PRINT "     ANTES DO MEIO D
IA     "
 210 FOR H=M TO N STEP 15*S
 220 LET I=INT (H/900)
 225 LET J=INT (.5+CODE "W"*(H/9
00-I))
 230 LET I=I+12
 235 IF I>24 THEN LET I=I-24
 240 LET K=PI+PI-PI*H/10800
 245 IF ABS (K)<>3*PI/2 AND ABS
(K)<>PI/2 THEN GOTO 250
 246 LET K=90
 247 GOTO 290
 250 LET K=180*(ATN (G*TAN K))/P
I
 260 LET K=ABS (INT (10*K+.5)/10
)
 270 IF H<=5400 OR H>=16200 THEN
 GOTO 290
 280 LET K=180-K
 290 IF K=PI-PI THEN LET I$=" "
 291 PRINT "H=>";I;TAB 6;"MIN=>"
;J;TAB 14;"ALFA=>";I$;K
 300 IF M=PI-PI THEN LET I$="+"
 310 NEXT H
 320 IF M=PI-PI THEN STOP
 322 PRINT
 325 PRINT "     APOS O MEIO DIA
     "
 340 LET M=PI-PI
 350 LET N=VAL "6300"
 360 GOTO VAL "210"
 400 PRINT AT CODE "■",PI+PI;"AL
FA E' SEMPRE NULO."

MEMORIA=> 1313 BYTES
```

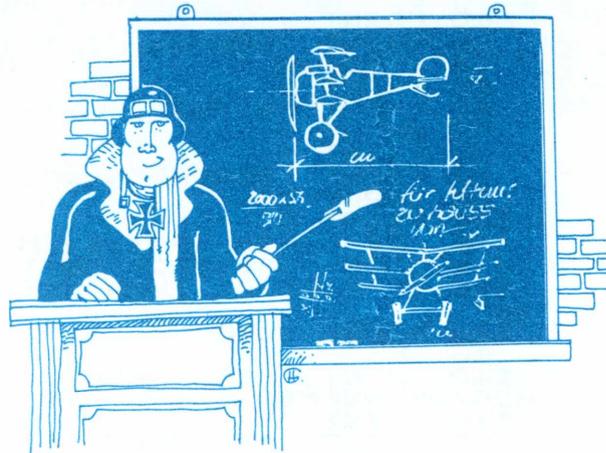
Respostas do Curso de BASIC

Aula 11

EXERC. 1

```
  2 LET A=1
  3 LET R1=1
  5 LET L=3
 10 LET R=1
 15 PRINT "MMC E MDC PARA NUMER
OS INTEIROS"
 20 GOSUB 1000
 25 LET N1=N
 30 GOSUB 1000
 35 LET N2=N
 40 LET F=1
 45 LET F=F+1
 50 IF F>3 THEN LET F=F+1
 52 IF N1<=1 THEN GOTO 58
 55 IF N1/F<>INT (N1/F) THEN GO
TO 45
 58 IF N2<=1 THEN GOTO 120
 60 IF N2/F<>INT (N2/F) THEN GO
TO 45
 62 PRINT AT L,0;F
 65 IF F<>A THEN LET R1=R1*F
 66 LET A=F
 69 LET R=R*F
 70 LET L=L+1
 80 LET N1=N1/F
 85 LET N2=N2/F
100 GOTO 40
120 PRINT AT 17,0;"M.M.C.= ";R
125 PRINT AT 18,0;"M.D.C.= ";R1
130 STOP
1000 PRINT AT 20,0;"DIGITE DOIS
NUMEROS (1.<=2.)"
1010 INPUT N
1020 PRINT AT L,0;"NUMERO= ";N
1030 LET L=L+1
1040 RETURN
```

EXERC. 2

```
 10 PRINT AT 20,0;"DIGITE N. LI
NHAS DE A"
 20 INPUT M
 30 PRINT AT 0,0;"LINHAS DA MAT
RIZ A= ";M
 40 PRINT AT 20,0;"DIGITE N. CO
LUNAS DE A"
 50 INPUT N
 60 PRINT AT 1,0;"COLUNAS DA MA
TRIZ A= ";N
 70 PRINT AT 20,0;"DIGITE N. LI
NHAS DE B"
 80 INPUT O
 90 PRINT AT 3,0;"LINHAS DA MAT
RIZ.B= ";O
100 PRINT AT 20,0;"DIGITE N. CO.
LUNAS DE B"
110 INPUT P
120 PRINT AT 4,0;"COLUNAS DA MA
TRIZ B= ";P
130 IF N<>O THEN GOTO 1000
140 DIM A(M,N)
150 DIM B(O,P)
160 DIM P(M,P)
170 PRINT AT 6,0;"MATRIZ A"
180 PRINT AT 20,0;"ENTRE COM OS
ELEMENTOS DE A"
185 LET L=8
188 LET C=1
190 FOR I=1 TO M
200 FOR J=1 TO N
210 INPUT A(I,J)
220 PRINT AT L,C;A(I,J)
225 LET C=C+2
230 NEXT J
234 LET L=L+1
236 LET C=1
240 NEXT I
250 PRINT AT 6,11;"MATRIZ B"
260 LET C=11
270 LET L=8
280 FOR I=1 TO O
290 FOR J=1 TO P
300 INPUT B(I,J)
310 PRINT AT L,C;B(I,J)
320 LET C=C+2
330 NEXT J
340 LET L=L+1
350 LET C=11
360 NEXT I
370 REM CALCULO DO PRODUTO
372 PRINT AT 6,21;"MAT.PROD."
374 LET C=21
375 LET L=8
380 FOR I=1 TO M
382 FOR J=1 TO P
385 LET P(I,J)=0
387 FOR K=1 TO N
390 LET P(I,J)=P(I,J)+A(I,K)*B(
K,J)
392 NEXT K
395 PRINT AT L,C;P(I,J)
397 LET C=C+2
400 NEXT J
405 LET L=L+1
410 LET C=21
415 NEXT I
420 PRINT AT 20,0;"FIM DO PROGR
AMA                            "
425 STOP
1000 PRINT AT 20,0;"O PRODUTO DE
MATRIZES NAO E POSSIVEL"
1010 STOP
```

EXERC. 3

```
  2 REM ESTE PROGRAMA GERA A SE
RIE DE FIBONACCI
 10 PRINT 0
 12 SCROLL
 15 PRINT 1
 17 SCROLL
 20 LET N1=0
 25 LET N2=1
 30 LET P=N1+N2
 35 PRINT P
 38 SCROLL
 40 LET N1=N2
 45 LET N2=P
 50 GOTO 30
```

EXERC. 4

```
  5 LET K=0
 10 FOR X=14 TO 33
 15 FOR Y=10 TO 14
 20 PLOT X,Y
 25 NEXT Y
 30 NEXT X
 35 LET A$="█        █"
 40 PRINT AT 13,7;A$;AT 14,7;"█
██████████"
 45 FOR X=5 TO 13
 50 PRINT AT X,12;":"
 55 NEXT X
 60 FOR X=17 TO 21
 65 PRINT AT X,12;":"
 70 NEXT X
 75 PRINT AT 1,4;"SENTIDO DE RO.
TACAO: ------>"
 78 IF K=3 THEN STOP
 80 LET B$=" █      █  "
 85 LET C$="  █    █   "
 90 LET D$="   █  █    "
 92 LET E$="    ██     "
 93 LET F$="     █     "
 95 PRINT AT 13,7;B$
100 PAUSE 30
105 PRINT AT 13,7;C$
110 PAUSE 30
115 PRINT AT 13,7;D$
120 PAUSE 30
125 PRINT AT 13,7;E$
130 PAUSE 30
135 PRINT AT 13,7;F$
140 PAUSE 30
145 PRINT AT 13,7;E$
148 PAUSE 30
150 PRINT AT 13,7;D$
152 PAUSE 30
155 PRINT AT 13,7;C$
158 PAUSE 30
160 PRINT AT 13,7;B$
162 PAUSE 30
165 PRINT AT 13,7;A$
170 PAUSE 30
190 LET K=K+1
200 GOTO 78
```

LIVROS

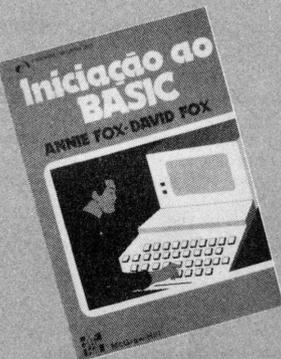
McGraw-Hill dá as dicas

Programação para TK-82, 83 e 85 e CP-200

Linda Hurley

É um livro dirigido às pessoas que estão começando a interessar-se por computação, principalmente àquelas que já tem um destes micros. Cada capítulo inicia-se com um programa curto e fácil. Porém, com o passar das páginas, o leitor já pode observar programas mais longos e inúmeras sugestões para mudá-los ou acrescentar neles suas próprias idéias, a medida que se vai acostumando com o micro.

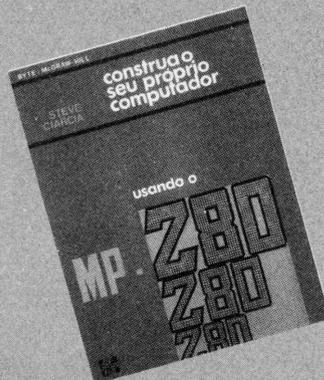
O livro possui 11 capítulos que abordam desde a classificação das teclas, o sistema de cores, palavras e desenhos até os movimentos, jogos e o último capítulo: Socorro! o que fazer? . . . *Programação* lembra bem uma cartilha do B a Bá pois as explicações são apresentadas de forma bem didática e colorida através de uma linguagem bastante acessível ao iniciante.

Iniciação ao BASIC

Annie Fox e David Fox

É um livro introdutório sobre a mais popular de todas as linguagens de programação de microcomputadores, dirigido aos principiantes. Dividido em 11 capítulos, *Iniciação ao BASIC* é formado de ilustrações, metáforas, exemplos e exercícios que atingem o intuito inicial de seus autores: apresentar os fundamentos da programação, aplicáveis a qualquer microcomputador que utilize a linguagem BASIC.

Os capítulos trazem a apresentação da computação aos leitores, através da explanação sobre os diversos momentos de sua história. Apresenta os equipamentos, explicando o hardware, o software e seus periféricos, as diversas instruções e comandos e qual a função de cada um; as variáveis e o que são números randômicos, etc.

No final de cada capítulo existe um pequeno questionário onde o leitor é indagado sobre os principais tópicos do capítulo. No final do livro o autor dá as respostas de todos os questionários e finaliza falando sobre o futuro da computação.

Construa o seu próprio computador usando o MP-Z80

Steve Garcia

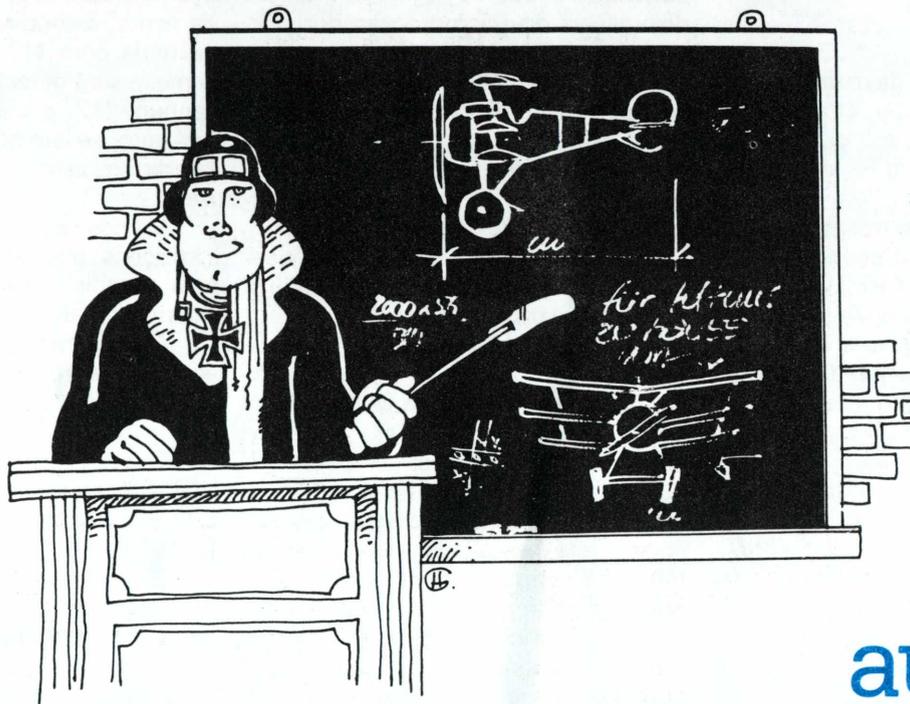
É um livro indicado aos mais afoitos. Àqueles que já passaram da fase inicial e posterior e agora possuem pretensões de maior porte: adicionar recursos maiores aos seus micros. O livro, com 400 páginas, é, na realidade, um guia prático que mostra como construir um computador baseado no famoso microprocessador — Zilog Z80. A descrição enfoca um microcomputador básico em placa única, contendo ZK de sistema operacional, portas serial e paralelas, display hexadecimal e armazenamento de massa em fita cassete.

Cada subsistema do computador é completamente explicado através de informações testadas que possibilitam aos leitores fazer mudanças no sistema, como por exemplo, um terminal de vídeo.

Processamento de Dados nas Empresas
Editora Atlas

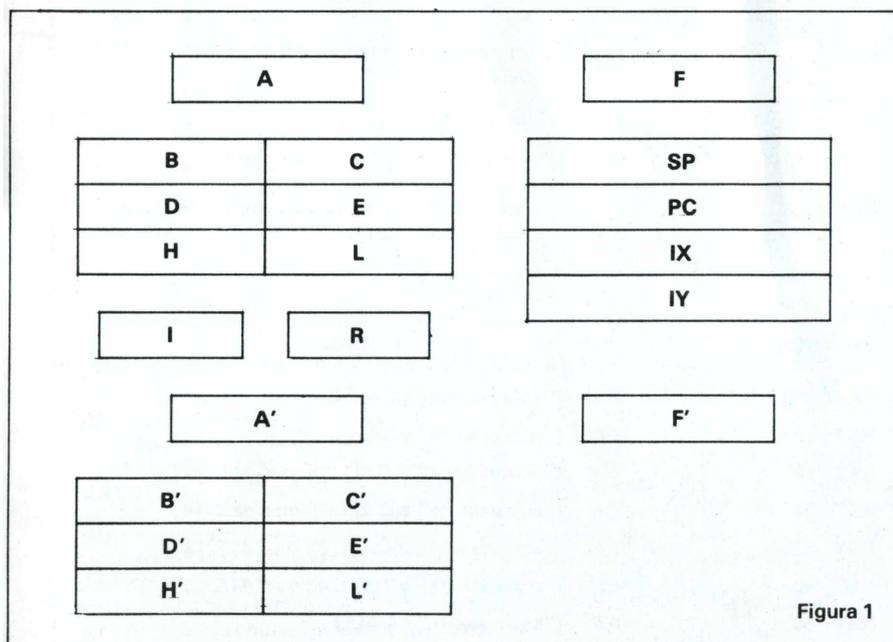
Tamio Shimizu

É um livro que deve constar na biblioteca de qualquer profissional da área. O autor, especialista na área de análise de sistema e de processamento de dados, incluiu no livro uma excelente coletânea de dados que possibilita ao leitor uma penetração segura na didática do processamento de dados. Segundo o próprio autor, no prefácio, o livro apresenta uma visão e treinamento gradual em PD na empresa, através do estudo integrado da máquina (o hardware), a análise e programação (software), a administração e a própria empresa.

CURSO DE ASSEMBLY

aula 12

Vamos, inicialmente, fazer um esquema indicando agora todos os registros inteiros do microprocessador; incluindo alguns novos que serão vistos nessa aula (figura 1).

Os registros A, B, C, D, E, H, L, F, A', B', C', D', E', H', L', F', I e R têm 8 bits cada e os registros SP, PC, IX e IY têm 16 bits cada. Os registros de 16 bits NÃO podem ser divididos em dois registros de 8; por outro lado, alguns registros de 8 bits podem ser "juntados" para formar registros de 16 bits; podemos então ter os pares: BC, HL, B'C', D'E' e H'L' e, em alguns casos, os pares AF e A'F'.

Como você deve lembrar, o registro SP é utilizado principalmente nas chamadas de sub-rotinas (para guardar o endereço de retorno na memória) e para "salvar" pares de registros; ao entrar em uma sub-rotina, é conveniente às vezes salvar todos os registros na "entrada" e obtê-los novamente na "saída" da sub-rotina; isto pode ser feito utilizando PUSHs e POPs. Este procedimento coloca os registros nas memórias da pilha indicadas por SP e, se quisermos salvar todos os registros, devemos fazer 4 PUSHs e 4 POPs! Entretanto existe um método mais fácil que economiza instruções e memória, usando a instrução:

EXX código '09' (exchange)

Ela utiliza os registros B', C', D', E', H' e L' para guardar os conteúdos de B, C, D, E, H e L de uma só vez; na realidade, ela *troca* os conteúdos dos registros um com o outro (B com B', C com C' etc.). Para "salvar" os registros A e F basta fazer:

EX AF, A'F' código '08'

que *troca* os conteúdos de A e F com A' e F'!

(NOTA: NUNCA use esta última instrução se você for operar em SLOW . . .).

Existem mais duas instruções de "trocas" de registros que você poderá utilizar:

EX DE, HL código 'EB'

Figura 1

que troca o conteúdo do registro D com H e do registro E com L.

EX (SP),HL código 'E3'

que troca o conteúdo da *memória indicada pelo Stack-pointer* (topo da pilha) com o registro L e o da *memória anterior* com o registro H, *sem alterar* o valor de SP.

Note que os registros A', B', C', D', E', H', L' e F' só são acessíveis através das instruções EXX e EX!

Vamos agora analisar os registros IX e IY; eles são registros de 16 bits cada, que podem ser utilizados *como o par de registros HL* (só que NÃO podem ser divididos em 2!) para endereçar memória e, além disso, permitem que uma constante seja *somada ou subtraída* ao endereço (desde que esta constante esteja entre – 128 e + 127); assim, por exemplo, se o registro IX for carregado com 30000:

```
LD      IX,30000
```

podemos fazer:

```
LD      B,(IX + 45)
```

que coloca no registro B o conteúdo da memória 30045! Para saber qual o código em hexadecimal das instruções que utilizam IX e IY, imagine que você esteja utilizando o par HL; a seguir, coloque na frente do código resultante o byte *'DD' se você for utilizar IX* ou *'FD' se você utilizar IY;* se IX ou IY estiver entre parênteses, então DEVE ser colocada uma constante, mesmo que ela seja zero! Esta constante, que sempre ocupa 1 byte (– 128 a 127) DEVE SER O TERCEIRO BYTE DO CÓDIGO OBRIGATORIAMENTE, mesmo que isto implique em dividir o código original em duas partes!

Assim, se o código de LD (HL), '24' é '3624', o código de LD (IX + '20'), '24' será 'DD362024'! Note como o nº '20' foi inserido de modo a ser o terceiro byte da instrução. O código de LD (IX), '24' seria 'DD360024'. Se o código de LD HL, 30000 é '213075' o código de LD YX, 30000 será 'FD213075'. Qual seria então o código de INC IX? E de INC (IY – 20)?

Estes registros são aqui apresentados apenas a título de complemento; apesar de serem mencionados neste curso, seu uso completo requer um estudo mais profundo da Linguagem de Máquina. O mesmo com relação aos registros I e R que não serão sequer mencionados neste curso, tendo sido colocados no desenho apenas para completar todos os registros internos disponíveis do microprocessador. Para um estudo mais completo do Assembly e seu uso no TK, você pode basear-se no livro de onde está sendo extraído esse curso (Introdução à Linguagem de Máquina para o TK – Assembly Z80" – vol. I e vol. II para aplicações mais complexas).

Os registros IX e IY podem ser usados *praticamente* em *todas* as instruções que utilizam o par HL.

NOTA: *NUNCA* tente utilizar o registro IX se você operar no modo SLOW.

De maneira análoga às instruções LDI, LDD, LDIR e LDDR, podemos usar a instrução CP para comparar "alguma coisa" com uma dada região de memória, para saber se esta "alguma coisa" está nessa dada região; assim temos as instruções da tabela I.

que utilizam o par BC como contador, o par HL como "ponteiro" para a memória e o conteúdo do acumulador contendo o que nós desejamos comparar; assim, por exemplo, para saber se na região de memória entre 30000 e 30099 existe 117, basta fazer:

```
LD      BC,100
LD      HL,30000
LD      A,117
CPIR
```

Feito isto, basta então verificar qual o conteúdo de HL para saber a posição em que está o número 117; obviamente, se houver mais do que uma memória com 117 nessa região, só a primeira será detectada; se não houver nenhum 117, o conteúdo final de BC será zero; se isto ocorrer, a flag Z é colocada em *zero;* caso contrário, será *um!*

Vamos agora complementar as instruções que veremos neste curso para depois partirmos para alguns exemplos mais "práticos".

Comecemos por uma que não faz absolutamente nada:

NOP código '00'

NOP é abreviação de NO OPERATION! Seu efeito é análogo, por exemplo, a LD D,D!

É difícil explicar agora a utilidade deste tipo de instrução . . . À medida que seus programas se tornarem mais complexos você perceberá que instruções como NOP poderão ajudá-lo: por exemplo, suponha que no MEIO de dado programa você introduza várias NOP em seguida. Se naquela região você mais tarde quiser substituir os NOP por outras instruções não terá problema; mas, se não houver os NOP e você quiser ACRESCENTAR instruções no meio de um programa, você

TABELA 1

INSTRUÇÃO	CÓDIGO	
CPI	'EDA1'	(compara A com memória apontada por HL, INC HL e DEC BC)
CPD	'EDA9'	(idem, mas DEC HL)
CPIR	'EDB1'	IDEM a CPI, mas repete até BC ser zero)
CPDR	'EDB9'	(idem a CPD, mas repete até BC ser zero)

TABELA 2

INSTRUÇÃO	CÓDIGO	
REST 0	'C7'	("equivalente" a CALL 0)
RST 8	'CF'	("equivalente" a CALL 8)
RST 16	'D7'	("equivalente" a CALL 16 ou CALL '10')
RST 24	'DF'	("equivalente" a CALL 24 ou CALL '18')
RST 32	'E7'	("equivalente" a CALL 32 ou CALL '20')
RST 40	'EF'	("equivalente" a CALL 40 ou CALL '28')
RST 48	'F7'	("equivalente" a CALL)
RST 56	'FF"	("equivalente" a CALL 56 ou CALL '38')

terá que *reescrever* todo o programa a partir do ponto em que você quiser a modificação. Outra utilização para o NOP é a "retirada" de instruções: basta substituir na memória a instrução por tantos NOP quantos forem os bytes ocupados pela instrução. Obviamente, colocaremos a partir do endereço inicial da instrução.

A seguir, temos a instrução RST que tem o mesmo efeito que CALL, mas ocupa apenas 1 byte! No entanto ela é restrita, pois NÃO possibilita chamadas *condicionais* de sub-rotinas (assim, por exemplo, não podemos fazer RST NZ) e ela *só permite acesso aos endereços 0, 8, 16 ('10'), 24 ('18'), 32 ('20'), 40 ('28'), 48 ('30') e 56 ('38').* (Tabela II)

Note que os endereços "chamados" pela instrução RST estão no começo da memória e portanto fazem parte da ROM. De fato, alguns desses endereços são a "entrada" de sub-rotinas do programa interpretador que poderão eventualmente ser utilizadas nos nossos futuros programas. Em aulas futuras daremos alguns exemplos do uso de RST. Existem ainda outras instruções que são usadas pelo microprocessador para "tratamento" de dispositivos "externos" (vídeo, teclado, impressora, gravador etc.) cuja utilização (e explicação) foge dos objetivos de um curso introdutório como este. São eles.

Finalmente, temos três instruções que "facilitam" o uso de números na base 10.

DAA RLD RRD

Labels

Para finalizar a aula, vamos apresentar o conceito de LABELS. Os LABELS são nomes que associamos às instruções em Linguagem de Máquina, ou melhor, a seus ENDEREÇOS, para facilitar a escrita do programa numa "primeira passada"; assim, todos os "cálculos de endereços" necessários às instruções de salto ou chamadas de sub-rotinas podem ser calculadas posteriormente. Observe o exemplo da figura 2.

```
LOOP  ADD       A,C
      INC       C
      LD        (HL),A
      DEC       B
      JR    NZ  LOOP
      CALL      NADA
      JP        LOOP
NADA  NOP
      NOP
      RET
```

Figura 2

Neste exemplo, LOOP e NADA são LABELS. Suponha agora que você queira colocar os códigos na memória e que o programa comece no endereço 30000; os LABELS que estão à esquerda são então calculados:

LOOP = 30000
NADA = 30012

(veja como é importante saber quantos bytes tem cada instrução, para poder calcular os endereços que correspondem aos LABELS).

A seguir, nos saltos *absolutos* e chamadas de sub-rotinas, eles podem ser substituídos diretamente:

CALL NADA
(CALL 30012 código 'CD3C75') 30012 = '753C'
JP LOOP
(JP 30000 código 'C33075') 30000 = '7530'

e, nos saltos *relativos,* um pequeno cálculo é necessário:

JR NZ LOOP
(JR NZ – 6 código '20FA') – 6 = 'FA'

Este método facilita muito a escrita de um programa em Linguagem de Máquina, evitando confusões desnecessárias por deixar todos os cálculos de endereços para o final, deixando então sua atenção *voltada* para a lógica do programa sem "distraí-la(o)" com cálculos de endereços. Além disso, os LABELS (aliados aos comentários) facilitam a compreensão do programa.

Exercícios

```
LD    D,(BC)
LD    A,(IX + 16)
ADD   HL,BC
EXX   BC, DE
SUB   HL,DE
LD    (15095),A
LD    (37040),22
AND   (IX + '20)
OR    (IY + 16)
XOR   HL
INC   IX
LD    B,(HL)
LD    BC,(30000)
SBC   A,A
SBC   HL,BC
SBC   IX,DE
```

Figura 3

Dada a lista de instruções da figura 3, indique quais delas existem e quais não existem. No caso de elas existirem, indique o número de bytes e o de flags afetados. ○

HOBBYSHOP VEJA SE SUA CIDADE TEM O QUE VOCÊ PRECISA

PARTNER

TECNOLOGIA NACIONAL DE VANGUARDA

A indústria brasileira de informática tenta hoje alcançar sua mais ambiciosa meta: a criação de tecnologia nacional de vanguarda.

Esta é também a principal bandeira da Elebra Informática, que em apenas 5 anos de atividades tornou-se a mais importante empresa fabricante de periféricos para computadores do país.

Suas unidades de disco e impressoras são concebidas e produzidas segundo rígidas normas de qualidade, sendo equiparáveis em custo e desempenho aos periféricos de maior sucesso do mercado internacional.

Através de um constante esforço em pesquisa e desenvolvimento, a Elebra Informática domina hoje a tecnologia de sua linha de produtos: a mais extensa do país.

Com isso seus equipamentos, além de perfeitamente adequados às necessidades do mercado nacional, têm a melhor relação custo/benefício, qualidade e performance, aliada a um alto grau de confiabilidade.

A Elebra Informática prova, assim, na prática, que a meta pode ser alcançada: com trabalho, investimento, mão-de-obra qualificada, e, sobretudo, com uma profunda crença no talento e capacidade da indústria brasileira em gerar tecnologia de vanguarda.

Filiada a ABICOMP

Elebra Informática S.A. - Avenida Brigadeiro Faria Lima, 1383 - 6º andar - Telefone: (011) 815.7633 - Telex (011) 25957 ELIN-BR - São Paulo - SP

Aqui você tem a melhor iniciação em microcomputação que existe.

O TK 83 já ensinou mais de 2 milhões de pessoas. Ele é muito fácil de operar. Usa o Basic, e a memória chega até 64 K bytes, e aceita monitor, impressora e joystick.

Num instante você vai estar resolvendo problemas programando, ou vencendo os muitos jogos disponíveis. O TK 83 não é só a melhor iniciação. Também é a mais divertida.

Aqui você já aplica os seus conhecimentos

Com o TK 85 você também pode se divertir muito: ele tem dezenas de jogos disponíveis.

Mas ele já é mais sofisticado. Tem software já pronto. Linguagens Basic e Assembler. Teclado tipo máquina de escrever, com 40 teclas e 160 funções. 16 ou 48 K de memória RAM, e 10 de ROM. Gravação em high-speed, e função Verify, para maior segurança.

Quando você já estiver apaixonado por microcomputação, ele vai corresponder totalmente.

Aqui você mostra tudo o que sabe.

O TK 2000 Color tem tudo que os melhores micros têm. Menos o preço. Aceita diskette, impressora (já vem com interface), alta resolução gráfica à cores podendo ser ligado ao seu TV colorido ou P&B. Tem 64 k de memória RAM e 16 k de memória ROM. Com excelente software disponível.

Você pode mostrar tudo o que sabe. Sem precisar mostrar muito dinheiro.

Preço sujeito a alteração

MICRODIGITAL

Microdigital Eletrônica Ltda
Caixa Postal 54121 - CEP 01000 -
São Paulo - SP Telex nº (011) 37008 MIDE BR

À venda nas boas casas do ramo, lojas especializadas de fotovideo-som e grandes magazines em: **ALAGOAS** - Maceió, Palmeira dos Indios, **AMAZONAS** - Manaus, **BAHIA** - Salvador, **CEARÁ** - Fortaleza, **DISTRITO FEDERAL** - Brasília, **ESPÍRITO SANTO** - Vitória, **GOIÁS** - Goiânia, **MATO GROSSO** - Cuiabá, **MINAS GERAIS** - Belo Horizonte, Divinópolis, Itajuba, Juiz de Fora, Poços de Caldas, São João Del Rei, Teófilo Otoni, Uberlândia, Uberaba, Viçosa, **PARAÍBA** - Campina Grande, **PARÁ** - Belém, **PARANÁ** - Curitiba, Londrina, Maringá, **PERNAMBUCO** - Recife, **RIO DE JANEIRO** - Campos, Niterói, Nova Friburgo, Petrópolis, Rezende, Rio de Janeiro, Volta Redonda, **RIO GRANDE DO SUL** - Bagé, Canoas, Caxias do Sul, Ijuí, Novo Hamburgo, Pelotas, Porto Alegre, Sant'Anna do Livramento, Santiago, Santa Rosa, São Leopoldo, **RIO GRANDE DO NORTE** - Natal, **RONDÔNIA** - Porto Velho, **SÃO PAULO** - Araraquara, Assis, Avaré, Bauru, Birigui, Botucatu, Campinas, Catanduva, Franca, Guarulhos, Itu, Jacarei, Jaú, Limeira, Lins, Marilia, Mogi Guaçu, Mogi das Cruzes, Ourinhos, Piracicaba, Pirassununga, Promissão, Rio Claro, Ribeirão Preto, Santos, Santa Barb. d'Oeste, São Bernardo do Campo, São João da Boa Vista, São Sebs. da Grama, São Carlos, São José do Rio Preto, São José dos Campos, Stº André, São Paulo, Sorocaba, Suzano, Taubaté, **SANTA CATARINA** - Blumenau, Brusque, Florinópolis, Itajaí, Joinville, **SERGIPE** - Aracajú, Se você não encontrar este equipamento na sua cidade ligue para (011) 800 - 255.8583.

MICROHOBBY

PEDIDO DE LIVROS E NÚMEROS ATRASADOS DA MICROHOBBY

(Preencher os ítens no verso e o cupom abaixo)

Sim, desejo receber os ítens assinalados no verso

NOME

ENDEREÇO

CEP

CIDADE

EST.

TOTAL DO PEDIDO: Cr$ ________________
Enviar cheque nominal cruzado ou vale postal
à MICROMEGA P.M.D. Ltda. — Cx. Postal 54096
CEP 01296 — São Paulo — SP
Cheque nº ________________
Bco. ________________ ☐ Vale postal

Estou ciente de que aguardarei em média
30 dias para o recebimento dos produtos
DATA ______ / ________ / ________

____
assinatura

CERTIFICADO ESPECIAL DE RESERVA
VÁLIDO ATÉ 31.10.84

AUTORIZO PELO PRESENTE MINHA ☐ ASSINATURA INICIAL ☐ RENOVAÇÃO ☐ ALTERAÇÃO DE ENDEREÇO DA REVISTA MICROHOBBY (12 EDIÇÕES).

ANEXO REMETO ☐ VALE POSTAL Nº ______________ ☐ CHEQUE Nº ______________ BCO. ______________

NOMINAL À MICROMEGA P.M.D. LTDA. NO VALOR DE CR$ ______________________

NOME ____

ENDEREÇO ____

CIDADE ______________ BAIRRO ______________ CEP ______________

FONE ______________________ ESTADO ______________________

PROMOÇÃO SENSACIONAL VOCÊ ECONOMIZA Cr$ 12.000/14.000

PREÇOS: ASSINATURA: Cr$ 22.000
RENOVAÇÃO: Cr$ 20.000

Em caso de renovação de assinatura colar neste campo a etiqueta de endereçamento atual.

Micromega P.M.D. Ltda. • Av. Angélica, 2318 • 14 And. • São Paulo • Cep 01296 • Caixa Postal 54096 • Fone: 255-0900